

SafeTI™ Compiler Qualification Kit

User's Guide



Literature Number: SPNU585D
July 2013—Revised November 2014

1	Scope of this Document.....	4
2	Glossary	5
3	Method: Model-Based Tool Qualification.....	6
4	Documentation Structure	7
5	Application of the Qualification Kit	10
5.1	Determination of Qualification Need	10
5.2	Qualification	10
5.3	Qualification Support Tool.....	11
5.3.1	Requirements	11
5.3.2	Installation	11
5.3.3	Starting the QST.....	12
5.3.4	Determination of Qualification Need and Error Detection.....	13
5.3.5	Qualification Planning.....	23
5.3.6	Generation of Qualification Documents	26
5.3.7	Finalization of the Tool Qualification Report	29
5.3.8	Customization.....	30
6	Extension of the Qualification Kit	31
7	References	32
A	Requirements Tracing to Safety Standards.....	33
A.1	Requirements of ISO 26262	33
A.2	Requirements of IEC 61508	35
A.3	Requirements of EN 50128	35
A.4	Requirements of DO-330 (Operation Parts)	35
A.5	Satisfaction of ISO 26262 Requirements	35
A.6	Satisfaction of IEC-61508 Requirements	36
A.7	Satisfaction of EN 50128	37
A.8	Satisfaction of Requirements of DO-330 (Operational Parts)	38
B	Revision History	39

List of Figures

4-1.	Derivation of Tool Safety Manual Contents	7
4-2.	Documentation Plan	9
5-1.	Qualification Support Tool.exe.....	11
5-2.	Welcome Screen after Start of QST	12
5-3.	Selection of Qualification Target Directory	12
5-4.	Help Information	13
5-5.	Selection of Safety Standard	14
5-6.	Selection of Variants	15
5-7.	Tools and Use-Case Selection	16
5-8.	Version Identification Selection.....	17
5-9.	New Version Dialog	17
5-10.	Known Bug Import Dialog	18
5-11.	Feature Selection Page	19
5-12.	All Feature Deselection.....	20
5-13.	Mitigation Selection Page (Default/Initial Configuration)	20
5-14.	Mitigation Selection Page (remaining errors)	21
5-15.	Mitigation Selection Page (all errors)	22
5-16.	Mitigation Selection for One Potential Error	23
5-17.	Qualification Planning - Role Assignment.....	24
5-18.	Qualification Planning - Step Planning	25
5-19.	Qualification Planning - Artifact Planning	26
5-20.	Qualification Summary with Paths to Generated Documents	27
5-21.	Finished Qualification Message	27
5-22.	Generated Documents Overview.....	28
5-23.	Customization of Document Properties	29
5-24.	Qualifications Kit in the QST	30
5-25.	Qualification Kit in the Qualification Target Directory.....	30
A-1.	ISO 26262 Tool Qualification Requirements	33

Scope of this Document

This document describes how to use the SafeTI™ Compiler Qualification Kit, in order to assist customers in qualifying the TI C/C++ compiler according to functional safety standards.

Since this qualification kit can be adapted to the process of the user and since it can be extended, there are other related documents that are included to support the user (see the documentation plan in [Chapter 4](#)). This document contains an overview of the qualification kit and explains how to use it and how to create the required documents to assist in the qualification of the TI C/C++ Compiler within your safety case.

The safety of tools is achieved within three steps:

1. Tool evaluation and possibly qualification
2. Proper tool installation
3. Proper tool operation

The qualification of the tool is achieved with the following steps:

1. Determination of the qualification need (from the tool analysis)
2. Creation of a tool qualification plan (in this case we use validation by test as main qualification method)
3. Execution of the tool qualification according to the test plan
4. Documentation of the tool qualification results in the tool qualification report and the tool safety manual.

Therefore this document is structured as follows:

- Method: Model-based tool qualification, see [Chapter 3](#)
- Documentation Structure, see [Chapter 4](#)
- Application of the qualification kit, see [Chapter 5](#)
- Extension of the qualification kit, see [Chapter 6](#)

Furthermore, the document demonstrates the standard compliance of the kit by tracing against the requirements from the relevant safety standards (see [Chapter 7](#)).

Glossary

This section defines technical terms used within this document. Note that elements, relations and actions from the model that have a formal semantic in the TCA are written in uppercase letters and with italic font, for example, “*Error* element”, or “*Export -> Excel Review*”.

Term	Definition
<i>Check</i>	To possibly detect an error
<i>Error</i>	In this document, used as “potential error”
Error (model) element	Representation of a (potential) error in the model
<i>Feature</i> (model) element	Representation of a function in the model
Function	An elementary or composed function of the tool that can be required in one or more use cases; for example, load, save, “perform” functions
Qualification environment	TAU and tests, a validation suite according to ISO 26262
<i>Restriction</i>	To possibly avoid an error
<i>Safety Guideline</i>	Guideline to mitigate some potential errors of the tool. Modeled as a <i>Check</i> or <i>Restriction</i> , either in an usual <i>Use Case</i> or <i>Feature</i> of the <i>Tool</i> , or in a separate, virtual <i>Feature</i> that can be required (added) by any use case of the same tool. Safety Guidelines are listed in the tool classification report.
software off-line support tool (IEC 6108)	According to IEC61508-4-3.2.11: software tool that supports a phase of the software development life cycle and that cannot directly influence the safety-related system during its run time.
TAU	Test Automation Unit: executes tests for the test suite
TD	Tool Error Detection (TD) probability for a potential error to be detected and avoided in a defined process TD1 – high detection probability TD2 – medium detection probability TD3 – low or unknown detection probability
TCL (ISO 26262-8)	Tool Confidence Level (ISO 26262): required confidence in the tool when used in the analyzed tool chain TCL1 – low confidence required TCL2 – medium confidence required TCL3 – high confidence required ⁽¹⁾
Test	Single test with result PASS/FAIL/ABORT
Test Directory	A directory containing one or more test (directories)
<i>Test</i> (model) element	Representation of a test directory in the model including a test description that specifies it
Test Suite	Structured set of single tests
Test Plan	List of test (directories) to be executed
Tool	A development tool according to ISO 26262
Tool Chain	A collection of tools, not necessarily forming an input/output chain
Tool classes (IEC 61508-4)	Software off-line support tools are classified into the following tool classes: T1 – generates no outputs which can directly or indirectly contribute to the executable code (including data) of the safety related system T2 – supports the test or verification of the design or executable code, where errors in the tool can fail to reveal defects but cannot directly create errors in the executable software T3 – generates outputs which can directly or indirectly contribute to the executable code of the safety related system
Tool Classification	Determination of the required tool confidence level (ISO26262: TCL or IEC 61508: tool classes)
Tool Evaluation	For tool criteria evaluation: see tool classification
Use-Case	The purpose of using the tool in development process
Use Case (model) element	Representation of n use-case in the model
<i>Virtual Feature</i>	A <i>Feature</i> is called virtual, if its virtual attribute is set to true. <i>Virtual Features</i> are modeled in a <i>Tool</i> , but are not implemented in the tool. They are used to model safety guidelines (documents) and can be added flexible as required features to use cases to denote that the use cases follow them. Virtual feature do not have errors.

⁽¹⁾ Of course, once the tools with TCL>1 have been qualified, the TCL can be regarded as existing tool confidence for the qualified ASIL rather than required tool confidence.

Method: Model-Based Tool Qualification

The tool qualification demonstrates that the tool is qualified to fulfill its task safely. Usually, this is achieved by fixing a reference process and a small set of reference use cases and demonstrating by tests that the tool is working correctly.

The problem with this approach is that in tool chains nowadays the tools are used in so many different ways, that a fixed reference process or use case cannot be considered if several qualifiable tools are integrated into one tool chain. Therefore, the tool qualification kit for the TI C/C++ Compiler is based on a model that can be adapted by the user in a flexible way such that the tool can also be qualified within a user-defined setting. The model-based tool qualification approach offers the following benefits:

- **Formalization:** The model is precise and decisions, for example, on assumptions, can be stored. Furthermore, it allows the user to express complex situations like alternative mitigations or variants in the tool chain in a clear way.
- **Flexibility:** The user can configure use-cases by selecting tool features and applicable mitigations. The user then has to run only the required qualification tests.
- **Reusability:** The tool model (including the error mitigations) can be reused in the use case definition. Furthermore the tool model can be combined with other tools to reduce the qualification need.
- **Consistency:** The model can be checked for consistency to avoid wrong assignments from mitigations to errors or to detect missing descriptions or deviations from the general error model.
- **Automation:** The modeling tool can determine the confidence required in the different use cases; it can help in the management of models (for example, using Excel interfaces, merging of models, ...), generation of reports, review checklists and test plans.
- **Analyzability:** The model can be used to analyze different situations with different variants of the tool chain, such as adding a new tool, or removing a check. Furthermore, the tool can analyze the costs of mitigations to find optimal solutions.

There are three processes involved in tool qualification:

1. Classification of the tool (within the process) and determination of the confidence needs
2. Tool Qualification, in this case by application of the qualification kit that validates the tool
3. Building or extending the qualification kit

This qualification kit supports all three processes. Classification and qualification are done by the Qualification Support Tool (QST) that is contained in this qualification kit, while extension can be done with the Tool Chain Analyzer (TCA) tool. The TCA tool can be freely downloaded from www.validas.de/TCA.html for Windows, Linux and MacOS, such that every user can use the tool easily.

Documentation Structure

The safety standards (ISO 26262, IEC 61508, DO-178/DO-330) require the user to analyze the tools used for the development of safety-critical products. The result of the analysis is a requirement on the reliability of the tool stated in the tool classification report.

The confidence is determined by an analysis of the use cases of the tool as used within the development process. If the tool has an impact on the safety of the product, all potential errors within the used features are analyzed for how they can be detected or avoided within the process. If there is no high probability for detecting or avoiding the errors, the tool has to be qualified to ensure the absence of these errors.

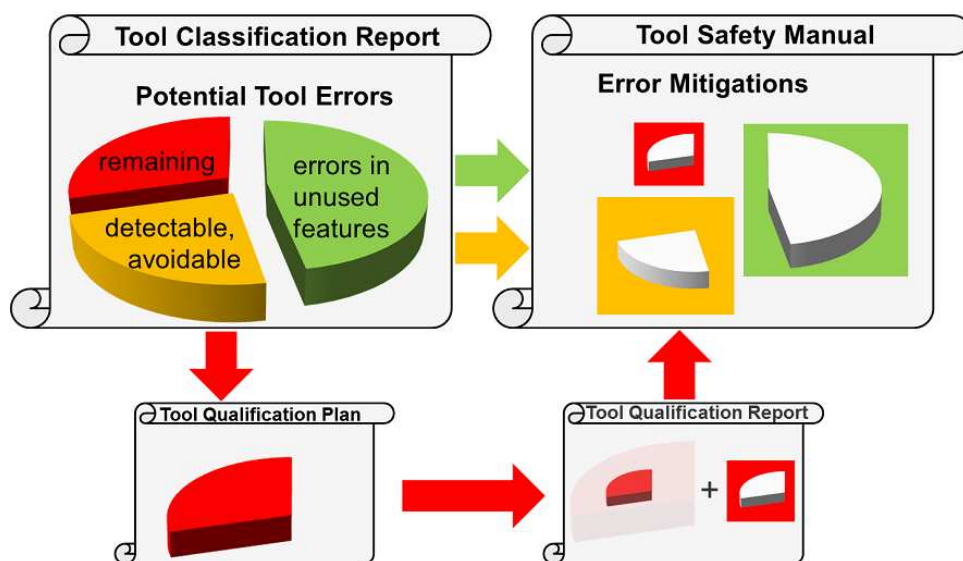


Figure 4-1. Derivation of Tool Safety Manual Contents

The tool safety manual for a tool has to contain the mitigations against all potential tool errors that are considered during tool evaluation [TCR]. The errors can be grouped into the three classes (see Figure 4-1):

- Potential errors in unused features (green in Figure 4-1). Using these features is prohibited in the tool safety manual.

NOTE: The analysis of potential errors in unused functions is not required, but the features need to be defined.

- Potential errors with mitigations: detections and restrictions (yellow in Figure 4-1): This includes potential errors for which detection and restriction mechanisms are described in the tool safety manual, especially if the checks and restrictions have to be triggered by the user of that tool.
- Remaining potential errors (red in Figure 4-1): Demonstrating the absence of this category of potential errors is the goal of the tool qualification (tool qualification plan). The tool qualification report (lower right chart in Figure 4-1) shows some specific evidence of errors that are instances of the potential error classes. The qualification report contains proposed workarounds for these specific errors that have to be part of the safety manual (together with the workaround for other already known relevant errors).

The tool safety manual therefore has to contain the following information:

- Allowed features and configurations of the tool
- Requirements to apply checks and restrictions to mitigate potential tool errors that might occur in required features and that are not excluded by tool qualification
- Workarounds for known errors and errors found during qualification
- Other information required by the standards to precisely identify the tool exactly (version, configuration, and so forth)

The tool qualification plan must identify potential errors of the TI C/C++ compiler that are not "detectable, avoidable" and show that they cannot occur. This is done by applying a validation suite in a systematic way that shows the absence of those potential errors that are not detectable, avoidable.

Since the TI C/C++ Compiler will be qualified using validation according to this qualification plan, the following documents are provided:

- *Test Plan*: specifies the required test cases for execution
- *Test Report*: contains the test results
- *TAU User Guide* [[TAG_UG](#)]: executes the planned tests cases correctly
- *Test suite validation and verification documents (plan and report)*: ensures that the test suite shows the absence of the potential errors if passed successfully

In the case that the model and the validation suite needs to be extended and new test cases need to be produced and validated, the following documents are required, or need to be extended:

- *Changed test specifications* including a test strategy to show the absence of the potential errors
- *Updated V and V plan and report* for changed tests

The test specification is part of the model (descriptions). The test suite needs validation against the potential errors of the model and verification against the implementation using a review. This quality process creates the confidence into the effectiveness of the test suite. The V and V documents for the test suite are contained in the qualification kit to demonstrate the confidence to the user. If the test suite is extended these documents will also be extended.

[Figure 4-2](#) shows the relationship between the documents and their variability which are constant and which depend on the use case. It describes how to derive the safety manual by a validation suite that consists of tests that show the absence of the identified critical errors in the tool evaluation report. Depending on the used features of the tool and the applied mitigation measures, this set of errors might vary. For every required test (or group of tests) that show the absence of one or more errors there needs to be a test specification (including a test strategy) that explains how the absence of the errors is ensured if the tests pass.

The tests in the test suite need to be validated to conform to the test specification. This is planned in a V and V plan of the kit and documented in the V and V report. Having a V and V report is the prerequisite for applying the validation suite to a use case. In [Figure 4-2](#) the user case-specific documents are in a green/inner, dashed box where the contents of the qualification kit are in the outer/blue box. Of course, the sequence of creating the documents (indicated by the sequence numbers) starts with the non user case-specific documents in the qualification kit. The tool qualification is planned in the qualification plan and requires executing tests (planned in the test plan) using the test automation unit manual [[TAU_UG](#)]. The test results are documented in a test report which is then analyzed and documented in the qualification report.

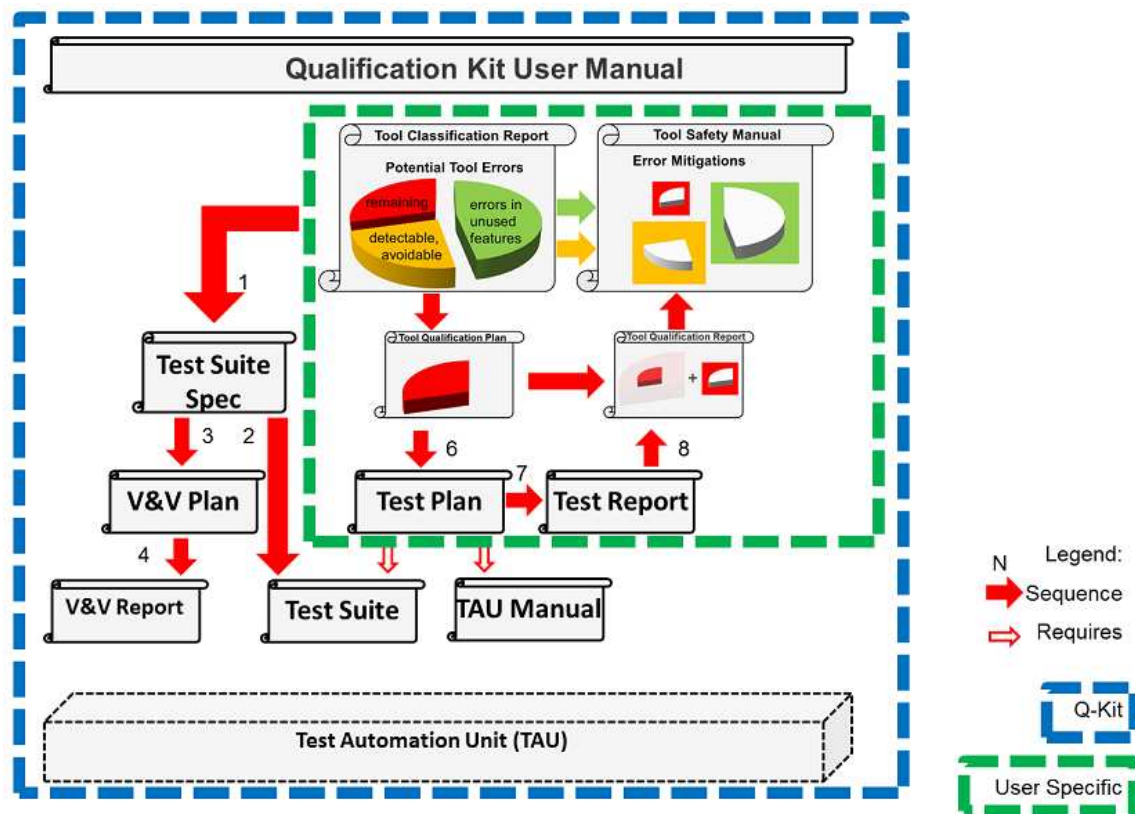


Figure 4-2. Documentation Plan

There are many documents in Figure 4-2 that are required and that need to be adapted depending on the user's process captured in the qualification model by selecting the required tool features and the executed mitigations during the process. The user case-specific parts in the user-specific documents are generated from the QST.

Application of the Qualification Kit

The tool qualification kit for the TI C/C++ Compiler is applied using the QST. This tool determines the qualification need for the TI C/C++ Compiler and, if required, prepares the qualification by testing. Furthermore, the tool generates the required documentation and artifacts.

This section describes:

- The determination of the qualification needs for the TI C/C++ Compiler (see [Section 5.1](#))
- The qualification of TI C/C++ Compiler (see [Section 5.2](#))
- The QST (see [Section 5.3](#))

5.1 Determination of Qualification Need

A tool requires qualification if it has potential errors that cannot be mitigated with a high probability. The potential errors depend on the features from the tool that are used. The error mitigation probability depends on the applied process, especially which checks and restrictions are applied in the use case of the tool, or within other tools.

The qualification need for TI C/C++ Compiler is determined by selecting the used features of the tool and by selecting the applied mitigations. While the list of features of TI C/C++ Compiler is constant, the list of mitigations depends on the selected features. For example, if no features are selected, then no mitigations are required.

Depending on the availability of test cases and possible mitigations, every feature (strictly speaking also every potential error) has one of these **qualification states**:

- Green: The feature is tested and can be used without usage constraints to mitigate potential errors
- Yellow: The feature cannot be tested and requires usage constraints
- Red: Neither test case nor error mitigations are available. The feature cannot be used without extending the qualification kit.

Whether the tool is qualified successfully depends on:

- The commitment to the required mitigations (if required by the selected features)
- The successful execution of the test cases (if required by the selected features)

The determination of the qualification needs can show either that test cases have to be executed or that usage constraints to mitigate potential tool errors have to be integrated into the processes (or a combination of both situations). The results of both are described in the tool safety manual that describes how the tool can be used safely.

5.2 Qualification

The qualification of the TI C/C++ Compiler depends on the qualification need. As described in [Section 5.1](#), the qualification depends on the selected features that require either test or mitigation to ensure the absence of potential errors. The results of both activities are integrated in to the tool safety manual.

In the case that test and mitigations are required, the qualification process consists of the following steps:

- Determination of qualification need, see [Section 5.1](#)
- Creation of documents (including a test plan)
- Execution of the tests with the TAU according to [\[TAU_UG\]](#)
 - Installation of the TAU
 - Execution of the required tests according the test plan

- Analysis of test results
- Finalization of documents

The QST (see [Section 5.3](#)) supports the determination of the qualification need and creates the documents such that the user only has to run tests and finalize documents.

5.3 Qualification Support Tool

The QST is part of the qualification kit. It supports the user during the qualification, by the determination of the qualification need and the generation of the qualification documents.

5.3.1 Requirements

The QST is available for Windows, Linux and MacOS-X systems and requires 1 GB memory. The tool does not require administration rights.

For the generation of the images in the documents the graphviz tool has to be installed and the (dot.exe) has to be in the execution path. Graphviz is open source and can be downloaded at <http://www.graphviz.org>. Graphviz version 2.28 or higher is required.

5.3.2 Installation

The QST is delivered as a zipped file. In the zipped file there is a directory that contains an executable that has to be used to start the qualification tool (see [Figure 5-1](#)).

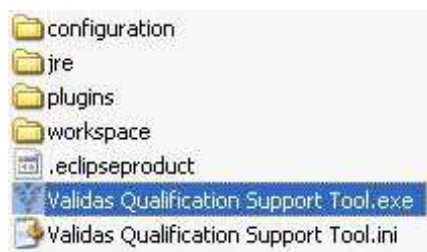


Figure 5-1. Qualification Support Tool.exe

5.3.3 Starting the QST

Double-click the executable to start the qualification tool. After starting the tool, the welcome screen appears as shown in [Figure 5-2](#).

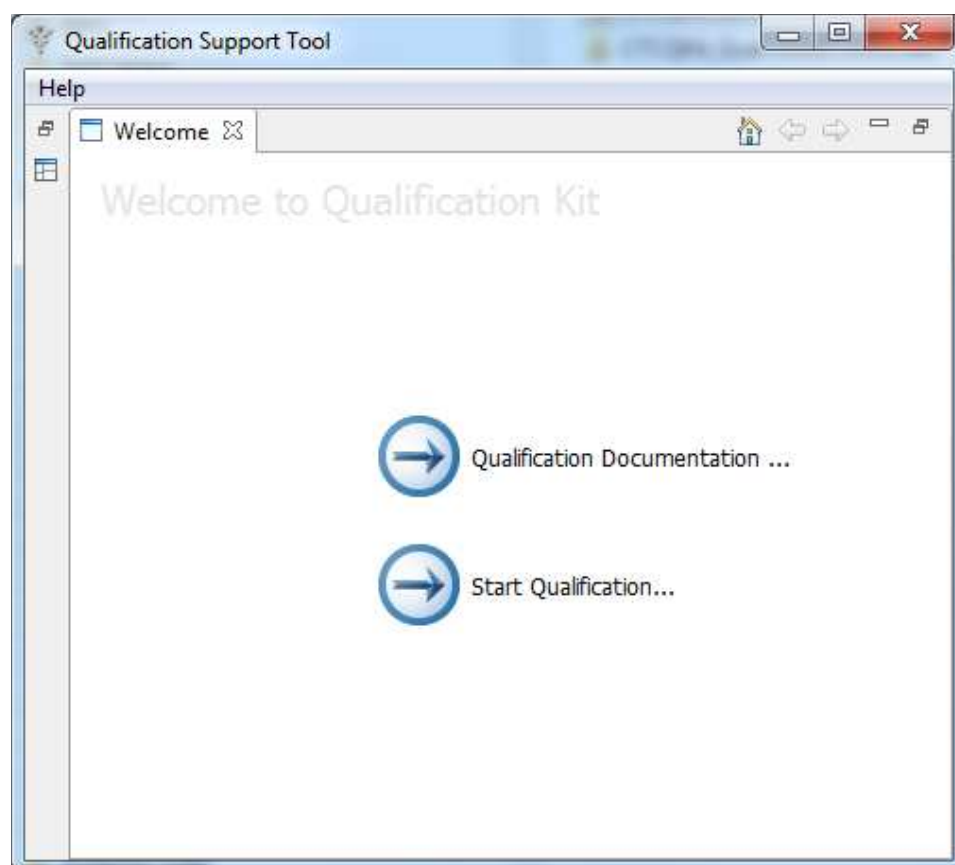


Figure 5-2. Welcome Screen after Start of QST

Click the “Start Qualification” button, and the qualification starts with the choice of a directory into which the qualification will be executed (see [Section 5.3.4](#)).

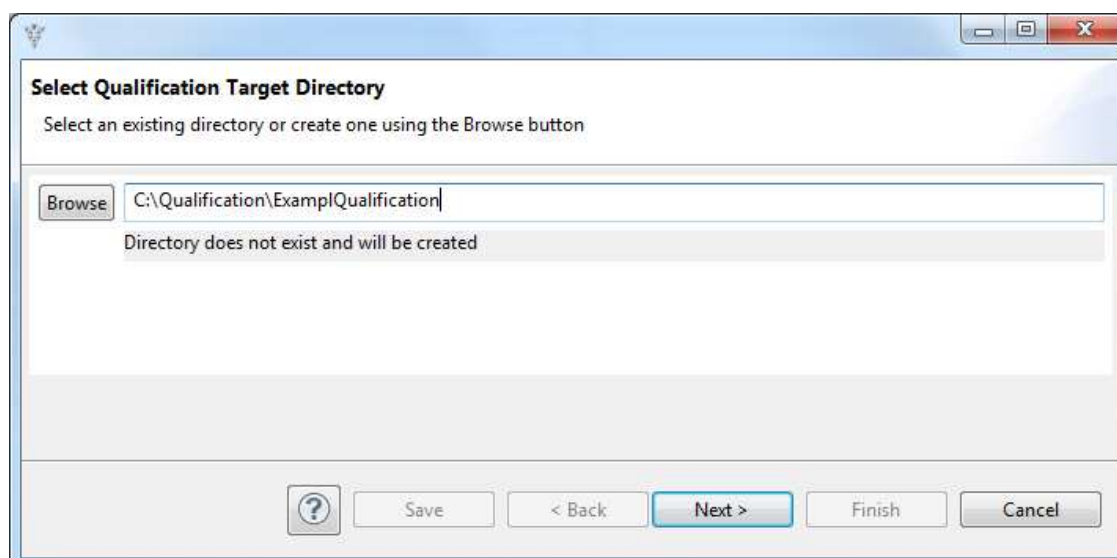


Figure 5-3. Selection of Qualification Target Directory

The qualification target directory is used to store all qualification artifacts. After the target directory is selected, the Qualification Support Tool can assist in the qualification need as described in [Section 5.3.4](#).

If the qualification target does not exist or is empty, a new qualification with the default settings from the QST and the model will be started. If the qualification target is not empty, the qualification will continue with the information (templates, model, status) found in the qualification target. This means that the QST can save the configuration status in the model and continue the qualification later.

If the Help Button(?) is pressed, a context-sensitive Help is displayed (see [Figure 5-4](#)).

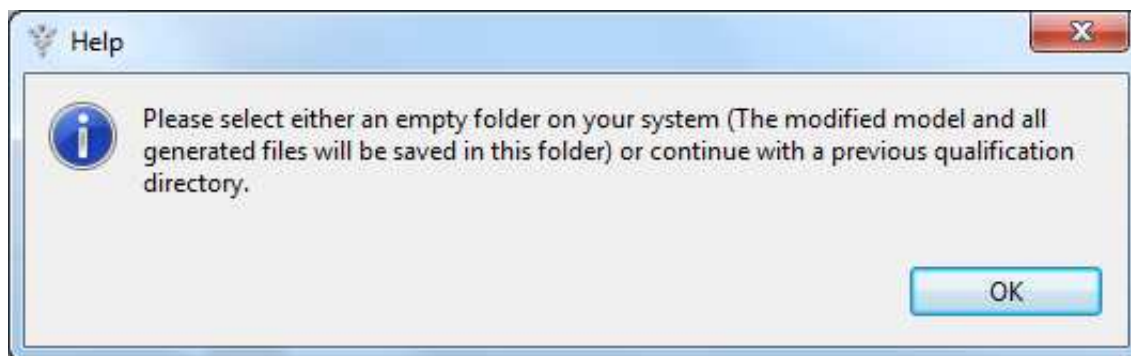


Figure 5-4. Help Information

NOTE: Do not forget to press the Save button if you want to continue with your selection; for example, to document your results using QST or to change some settings for a new qualification. The model is stored in the file <QualificationTarget>/Qkit/Model/Model.tca. If a qualification target is chosen that contains a saved model, this will be used as the default configuration.

5.3.4 Determination of Qualification Need and Error Detection

To determine the qualification need the QST asks for the standard according to which the tool will be classified and qualified (see [Figure 5-5](#)).

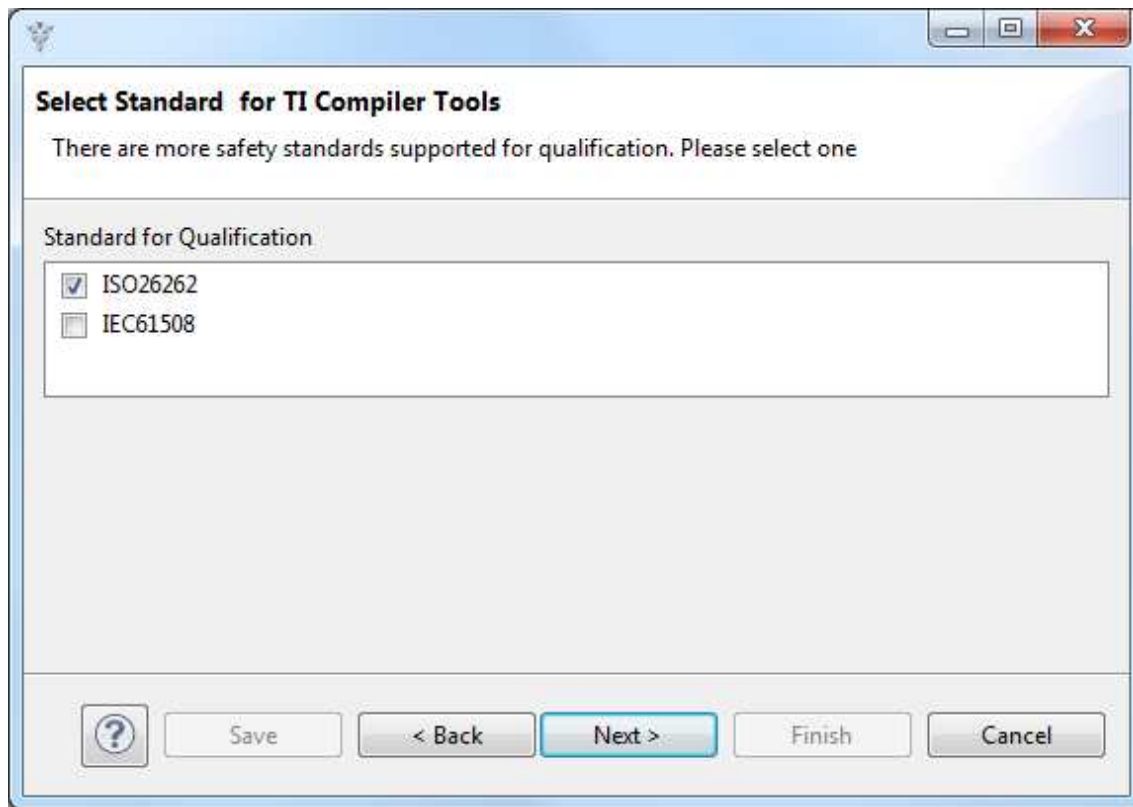


Figure 5-5. Selection of Safety Standard

The next step is to select the variant of the tool chain. This is only necessary if several variants are contained in the model. Most elements in the tool chain model can depend on variants, that is, are disabled if the variant is not chosen. [Figure 5-6](#) shows the selection dialog of variants that can be in the tool, for example for testing on the PC or testing on the target there might be different artifacts and features of the test tool used.

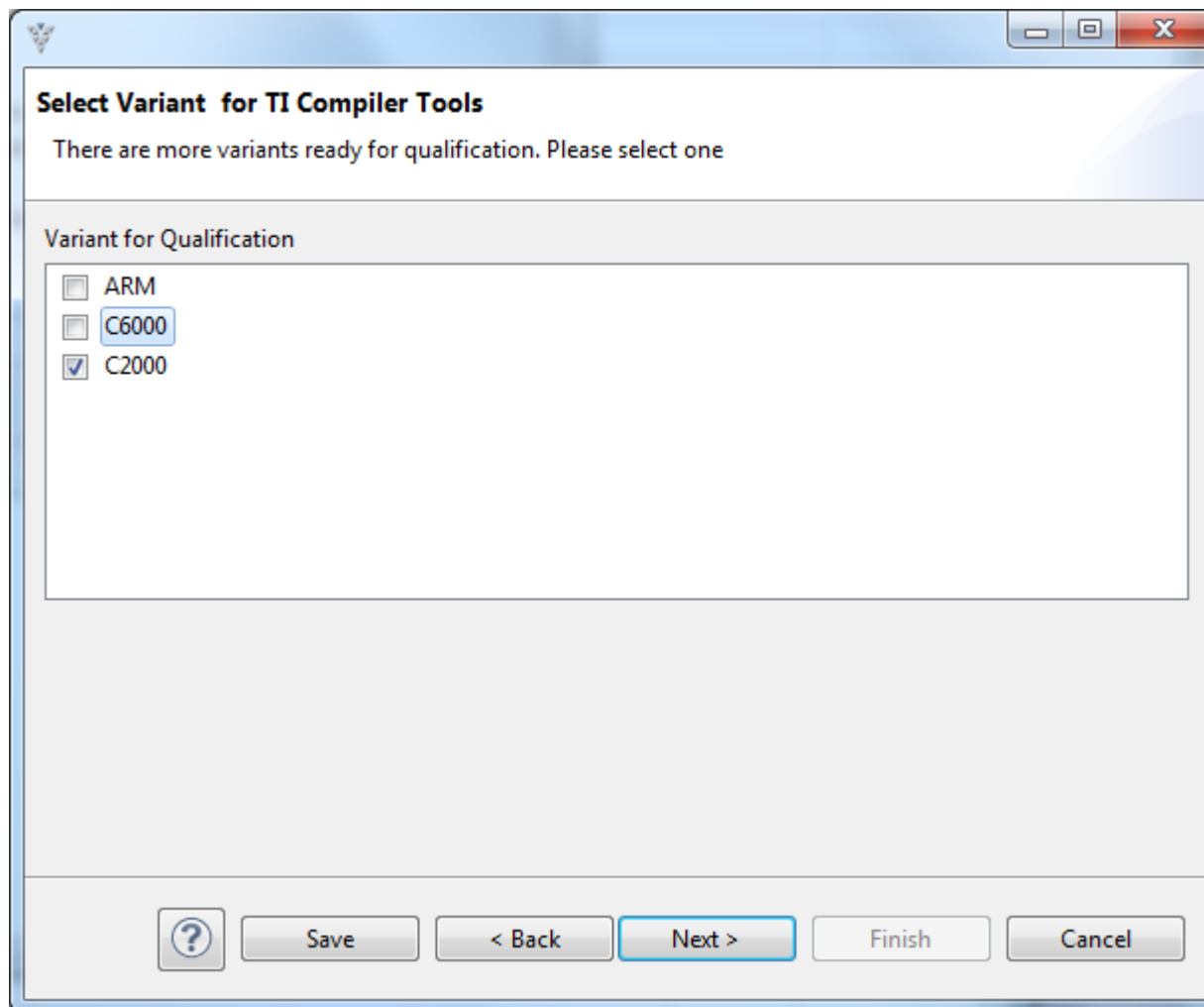
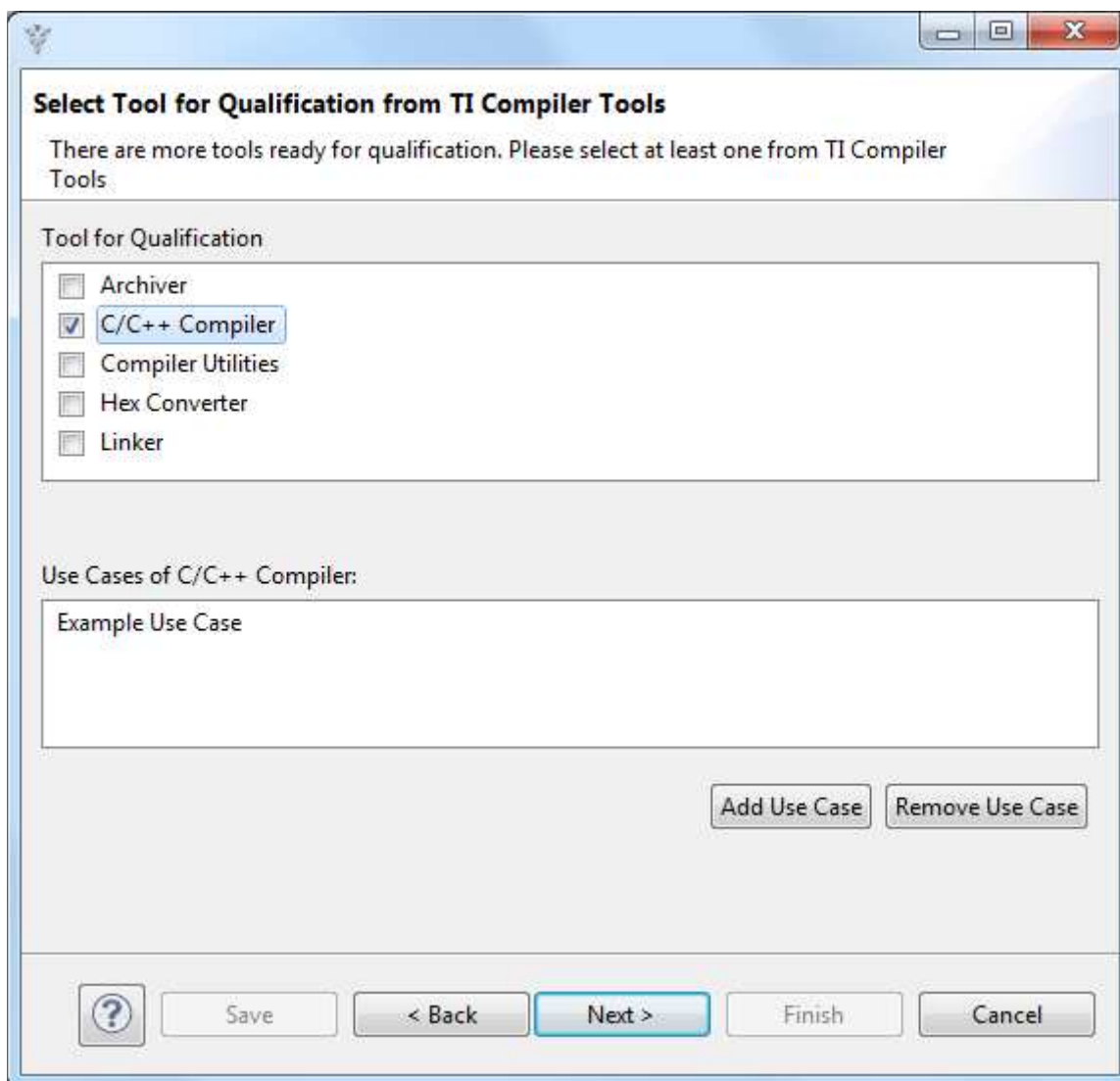


Figure 5-6. Selection of Variants

The next page is the selection of tools and use cases for qualification. Every use case of the tools will be configured and prepared for qualification. The following identification selection and use-case configuration page will be repeated for each use case that is defined in the tool and use case selection page, see [Figure 5-7](#). It shows the list of available tools that can be selected for qualification. Note that if the qualification need can be inferred according to the chosen standard from the model, it is depicted in brackets behind the tool name.



Select Tool for Qualification from TI Compiler Tools

There are more tools ready for qualification. Please select at least one from TI Compiler Tools

Tool for Qualification

- ☐ Archiver
- ☒ C/C++ Compiler
- ☐ Compiler Utilities
- ☐ Hex Converter
- ☐ Linker

Use Cases of C/C++ Compiler:

Example Use Case

Add Use Case Remove Use Case

? Save < Back Next > Finish Cancel

Figure 5-7. Tools and Use-Case Selection

The selection of the version/identification of the tool is the next step (see [Figure 5-8](#)).

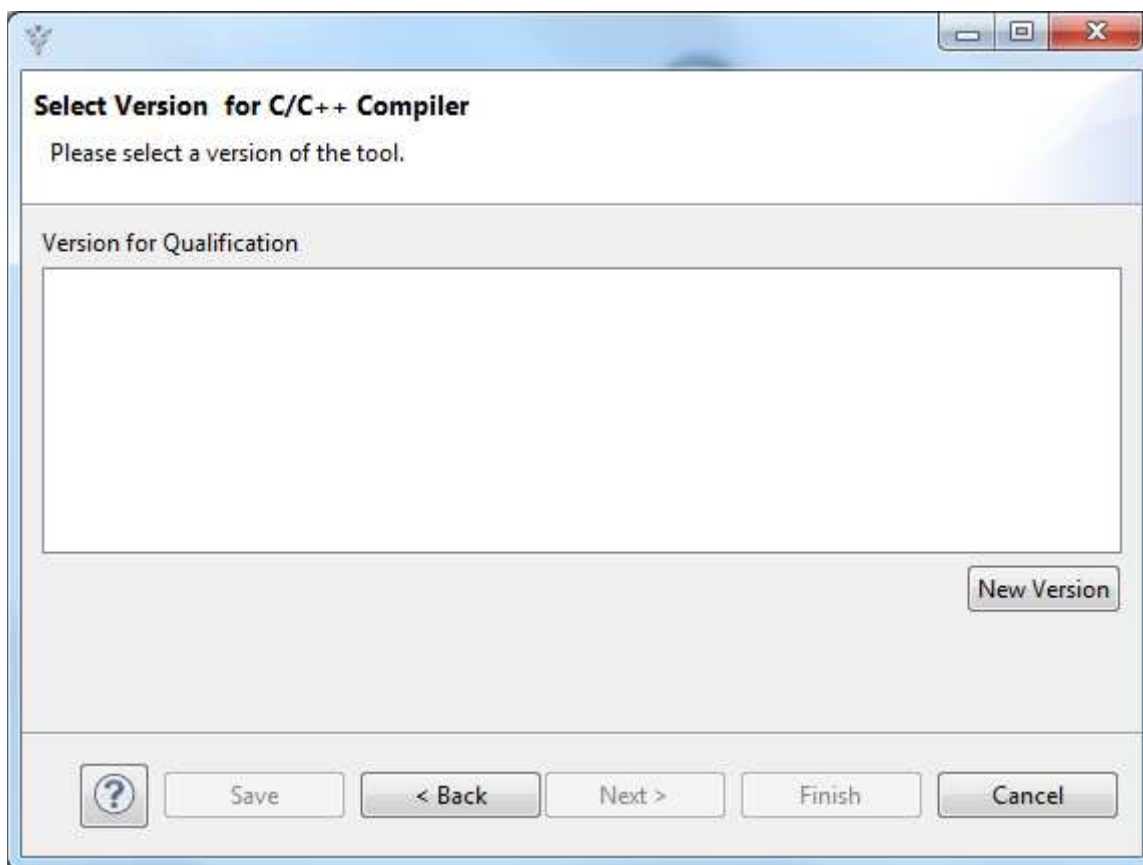


Figure 5-8. Version Identification Selection

New versions can also be added by pressing the new Version button, shown in [Figure 5-9](#).

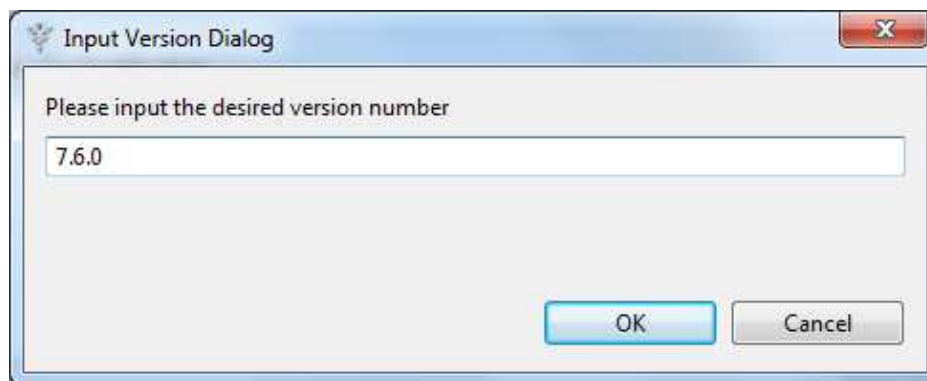


Figure 5-9. New Version Dialog

After the new version has been entered the user can add a file, for example a revision history file, containing the known bugs into the model (see [Figure 5-10](#)). The contained information will be added to the model and printed out in the generated tool safety manual.



Figure 5-10. Known Bug Import Dialog

The determination of the qualification need is performed in two steps:

- selection of the features for all tools with qualification need
- selection of mitigations or tests for potential errors in the selected features (also for all tools with qualification need)

The selection of the features for the tools is done in the feature selection page (see [Figure 5-11](#)). It shows cases of the tools (left side) and the available features of the tools (right side). The features have three different colors:

- 1. **green** colored features that can be used without constraints, since they are testable
- 2. **pink** colored features that can be used with some constraints (mitigations)
- 3. **red** colored features (if available)

The preselected features are those from the default model of the tool. In the bottom there is an information window that shows information.

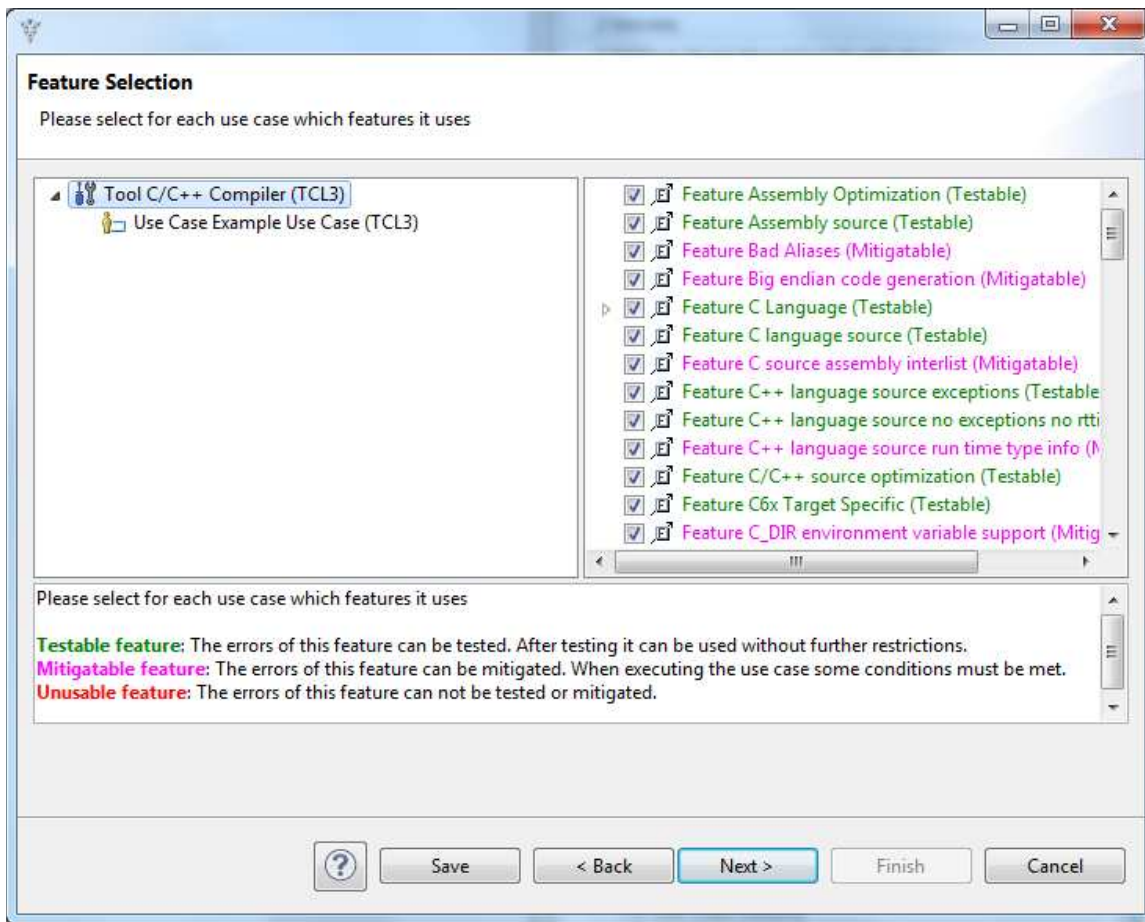


Figure 5-11. Feature Selection Page

- In the feature selection you can select the features or options that are used during your application of the tool.
- Select or deselect them by clicking by checking or unchecking the boxes in the right half of the dialog.
- Select or deselect all features with the right mouse button on the use cases in the left part of the dialog as shown in [Figure 5-12](#).

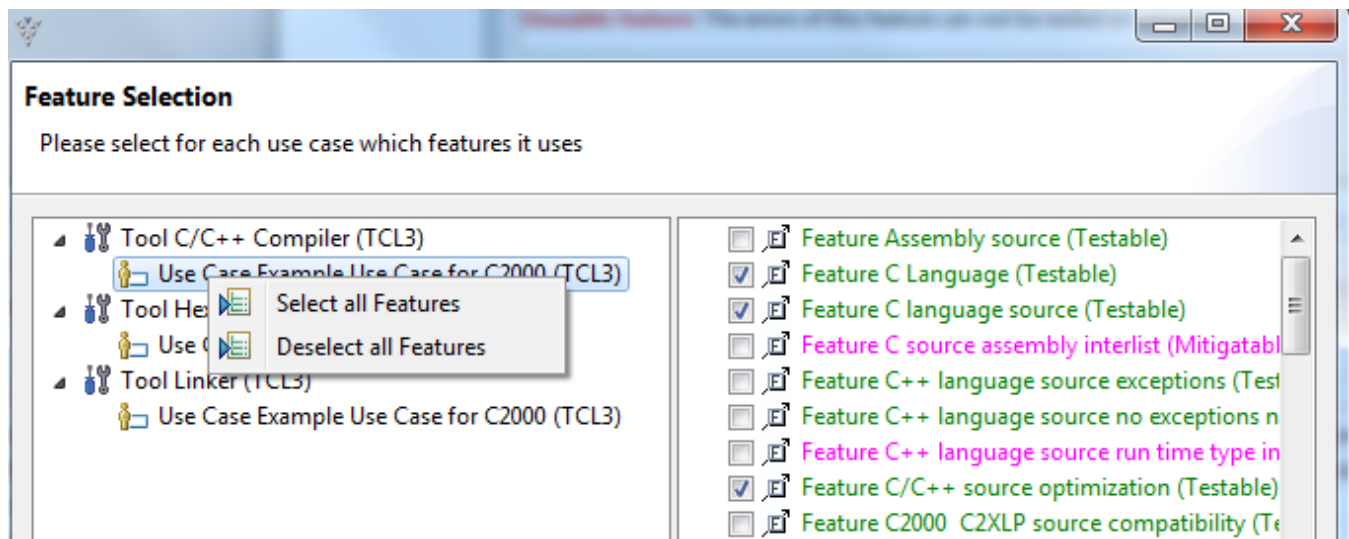


Figure 5-12. All Feature Deselection

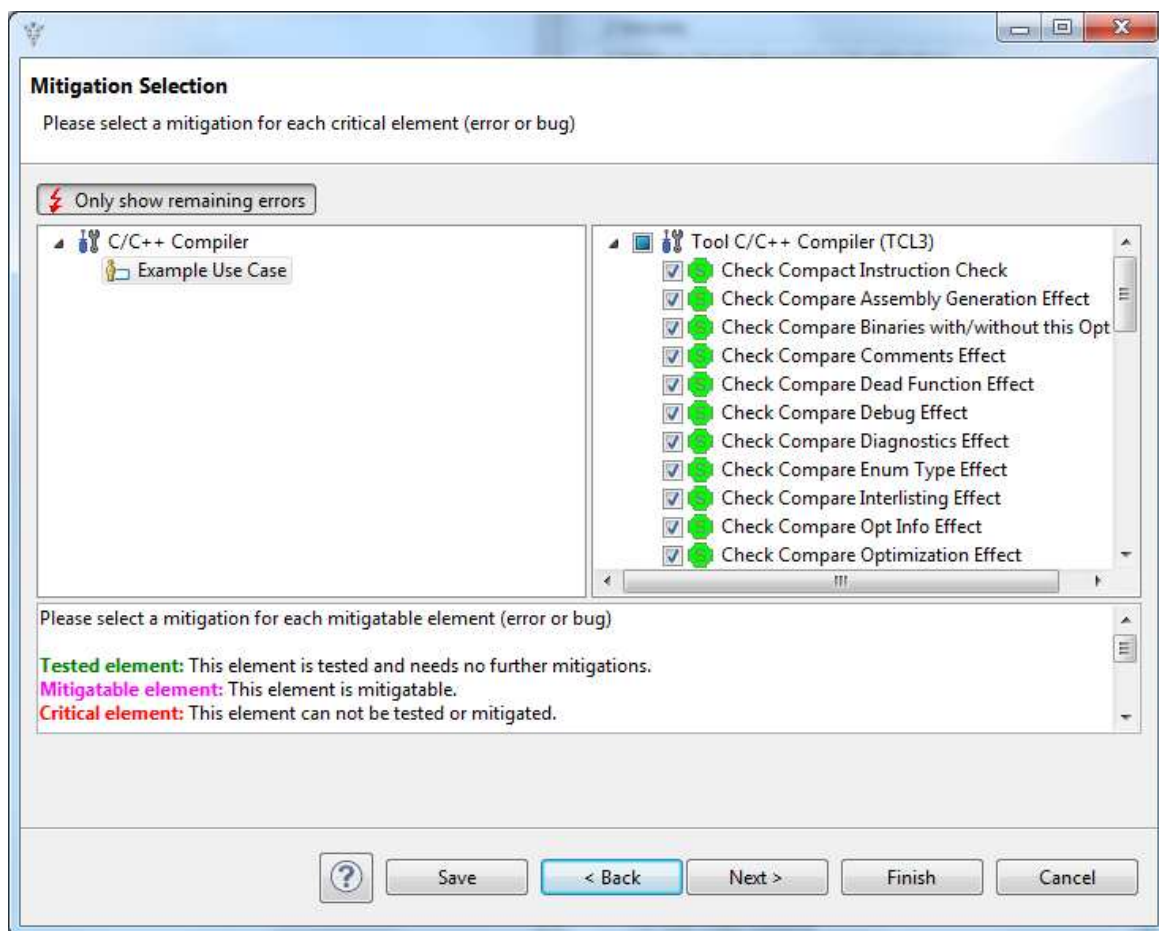


Figure 5-13. Mitigation Selection Page (Default/Initial Configuration)

The selection of the mitigations is the next step (see [Figure 5-14](#)). The page contains the list of potential errors on the left side. After selecting an error the available mitigations are shown on the right side and can be selected or deselected by setting/unsetting their check marks. If errors are mitigated they are marked as mitigated in the left side.

The default use cases in the model are so configured, that the initial mitigation page does not show any unmitigated errors, see [Figure 5-13](#). If additional mitigatable (pink) features which require new mitigations have been selected in the feature dialog this mitigation dialog will display the remaining errors as shown in [Figure 5-13](#).

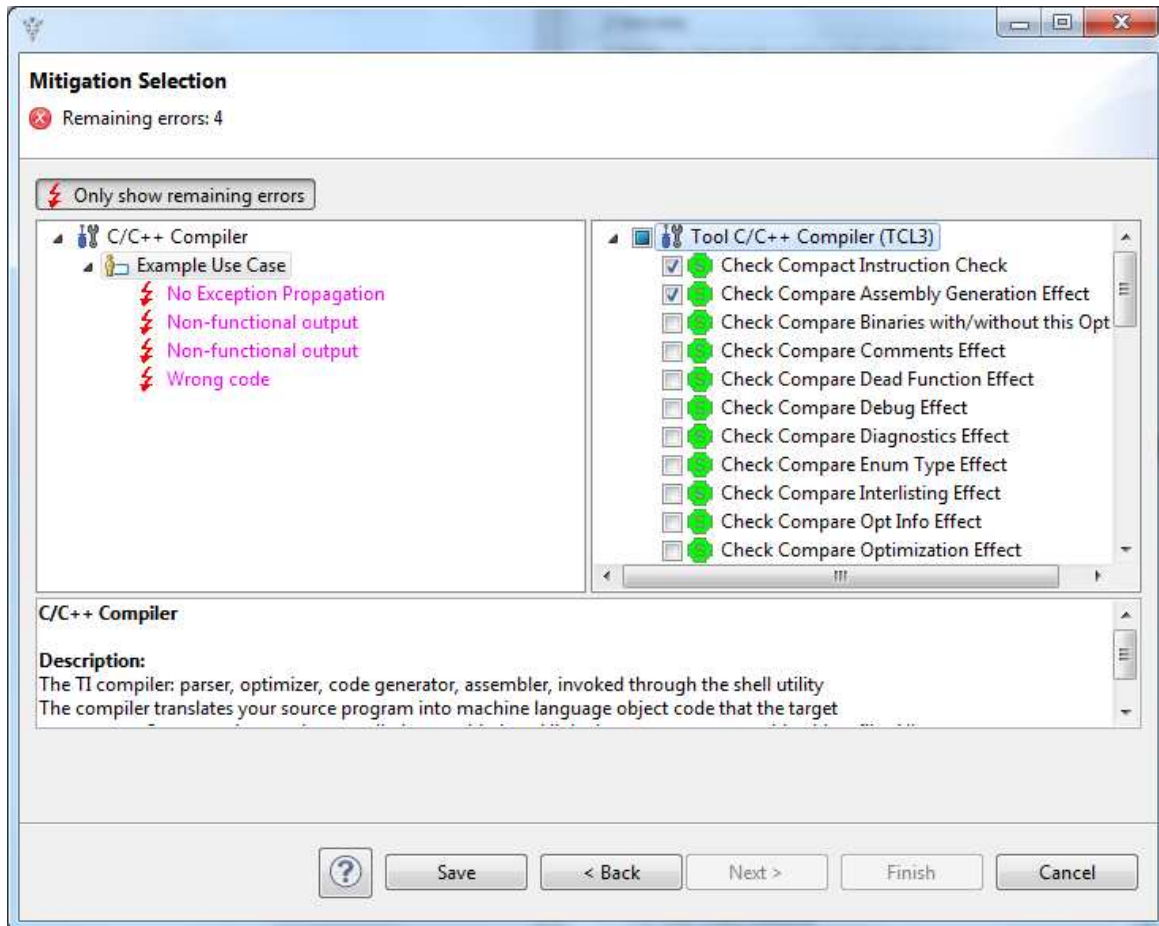


Figure 5-14. Mitigation Selection Page (remaining errors)

To see all potential and known errors (including the testable and mitigated) the user can toggle the button "Only show remaining errors" above the list. The QST will display all errors and their mitigations as shown in [Figure 5-16](#).

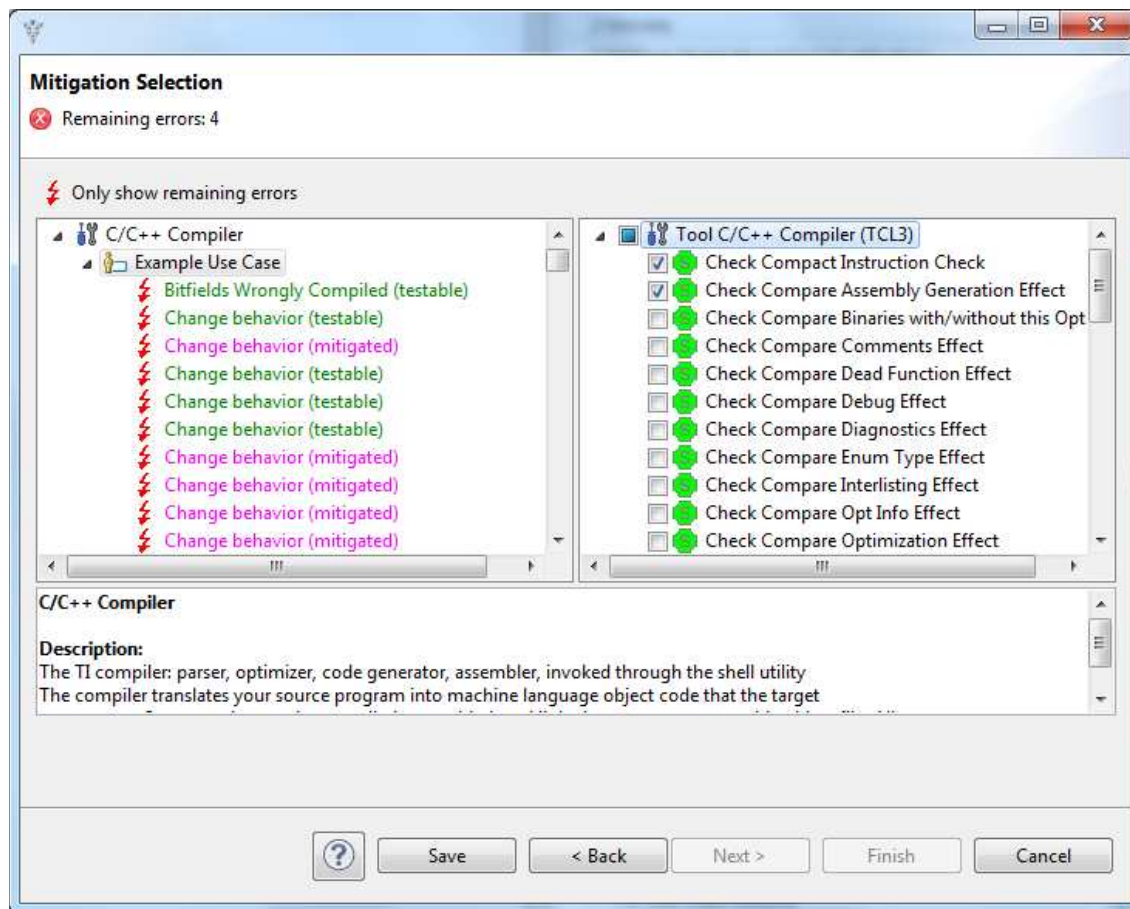


Figure 5-15. Mitigation Selection Page (all errors)

NOTE: You can also select mitigations specified in other tools, provided that there is a data flow from the tool with the error to and from the tool that detects or avoids the error. QST will show you all applicable mitigations when you select a potential error (see [Figure 5-16](#)). Typically, it is useful to select only one check to eliminate doing the work twice. If several tools are used, the checks will be performed with the last tool in the tool chain, in order to check the output of all tools.

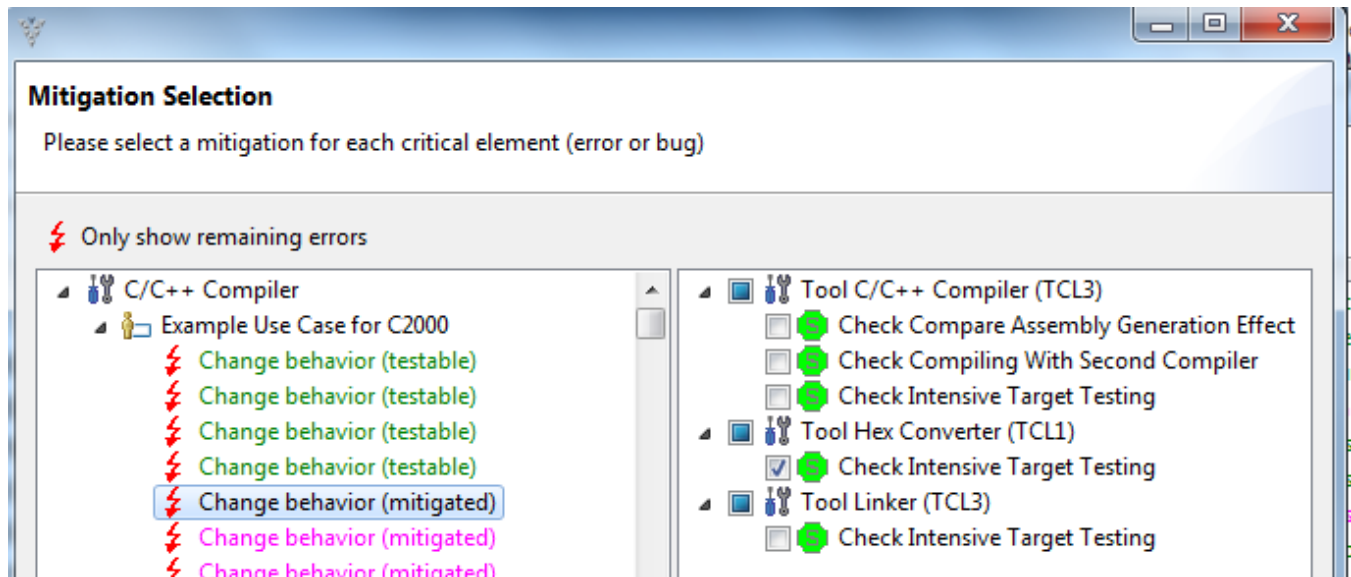


Figure 5-16. Mitigation Selection for One Potential Error

Navigation: At the bottom (below the information window) there is a navigation line, that allows to go to save the status (“Save”) the next page (“Next >”) or finish (“Finish”) the qualification preparation. These buttons are only enabled if sufficient mitigations are selected for all potential errors that have no test cases.

Note that the qualification kit comes with a predefined use case that is already qualifiable. In [Section 5.3.8](#) there are descriptions how this can be changed according to your selections to simplify the requalification with your selection.

5.3.5 Qualification Planning

The qualification mode contains a formalization of the qualification process the qualification can be planned using the QST. The plan will be generated into the Tool Qualification Plan as described in [Section 5.3.6](#). Tool qualification planning consists of three different elements that need to be specified:

1. Role assignment: there are qualification roles in the model, that need to be assigned to concrete persons that will be able to fulfill the role (see [Figure 5-17](#)). For that purpose they need to be selected in the tree and then the names can be edited.
2. Qualification Step Planning: allows to select the start and end date for the qualification steps and to assign responsible roles to them, see [Figure 5-18](#). Note that qualification steps can be hierarchic it is not necessary to plan each atomic step, but it is allowed to plan groups of steps. In this case the inherited information will be displayed.
3. Artifact Planning: allows the user to select the path for the artifacts, see [Figure 5-19](#).

Note that the planning will be updated during qualification, for example to update the finished dates or the paths to the produced artifacts. By creating this information the qualification plan can be extended to a qualification report that documents all steps of the project.

All planned information will be contained in the tool qualification plan, that is extended to the tool qualification report as described in the plan.

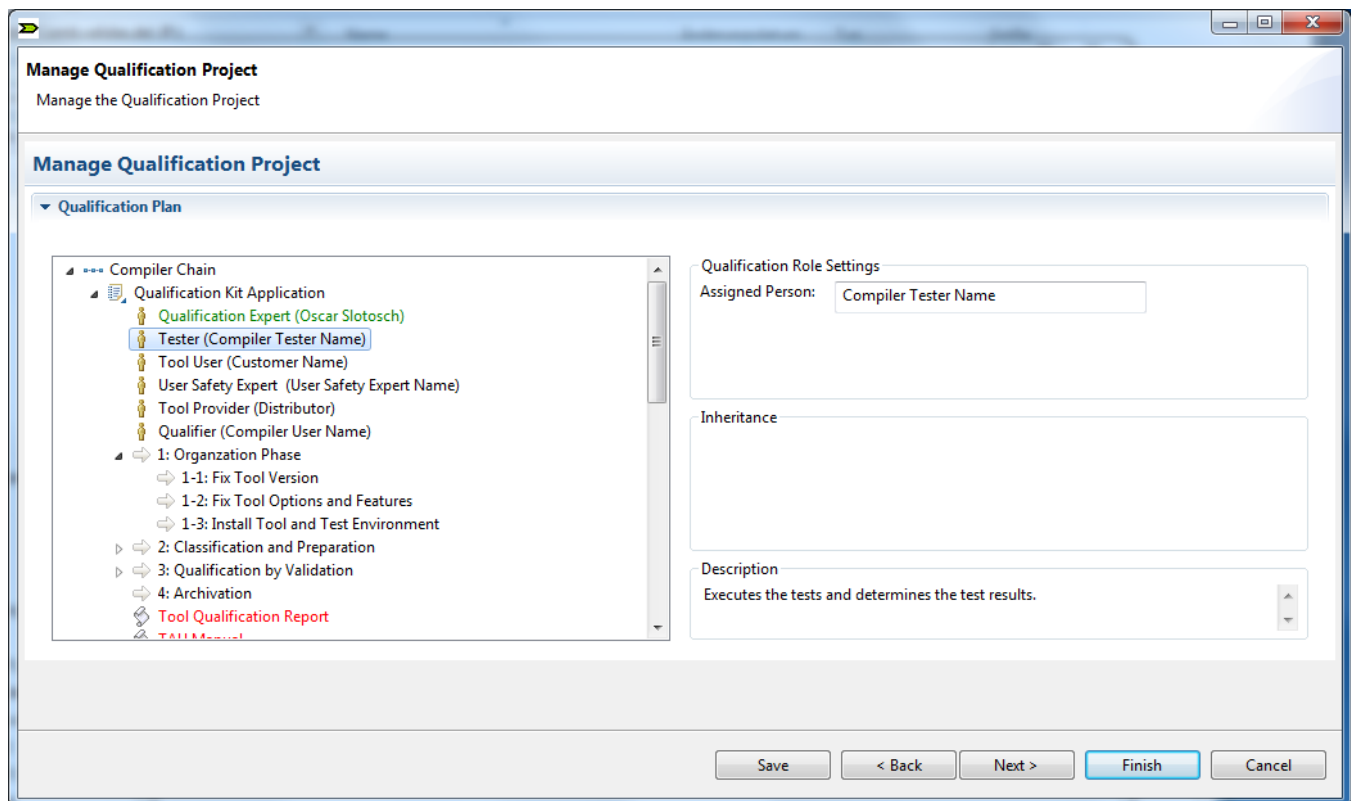


Figure 5-17. Qualification Planning - Role Assignment

Manage Qualification Project
Manage the Qualification Project

Manage Qualification Project

▼ Qualification Plan

- *** Compiler Chain
 - Qualification Kit Application
 - Qualification Expert (Oscar Slotosch)
 - Tester (Compiler Tester Name)
 - Tool User (Customer Name)
 - User Safety Expert (User Safety Expert Name)
 - Tool Provider (Distributor)
 - Qualifier (Compiler User Name)
 - 1: Organization Phase
 - 1-1: Fix Tool Version** (Selected)
 - 1-2: Fix Tool Options and Features
 - 1-3: Install Tool and Test Environment
 - 2: Classification and Preparation
 - 3: Qualification by Validation
 - 4: Archivation
 - Tool Qualification Report
 - TAU Manual
 - TAU
 - Tool Classification Report
 - Tool Safety Manual
 - Tool Qualification Plan
 - Coverage Measurement Report
 - Feature List
 - Test Report
 - Tool Definition
 - Model
 - Compiler, Uninstrumented
 - Compiler, Instrumented
 - Coverage Measurement Scripts

Qualification Step Settings

Planned Date: 17.10.2013

Finished Date: 17.10.2013

Qualification Role: Tool User

Inheritance

Planned Date: No date set in super steps

Finished Date: No date set in super steps

Qualification Role: No inheritance needed

Description

The tool version has to be fixed in order to produce an instrumented version of the compiler.
During this phase of the qualification project, the participants of the project agree on a fixed tool version, that has to be qualified. This is referred to as the tool definition.

Inputs:
Model

Outputs:
Model
Tool Definition

Figure 5-18. Qualification Planning - Step Planning

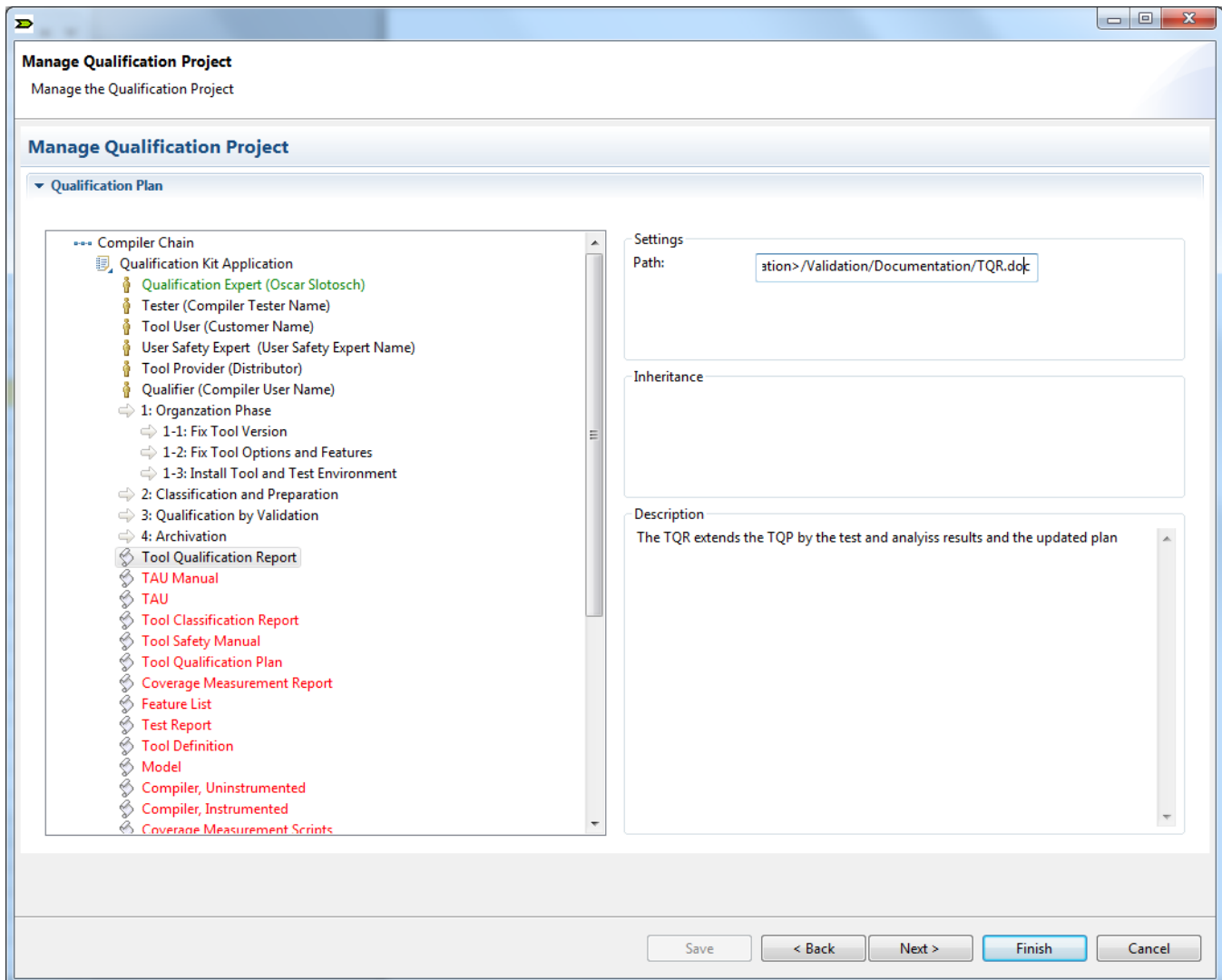


Figure 5-19. Qualification Planning - Artifact Planning

5.3.6 Generation of Qualification Documents

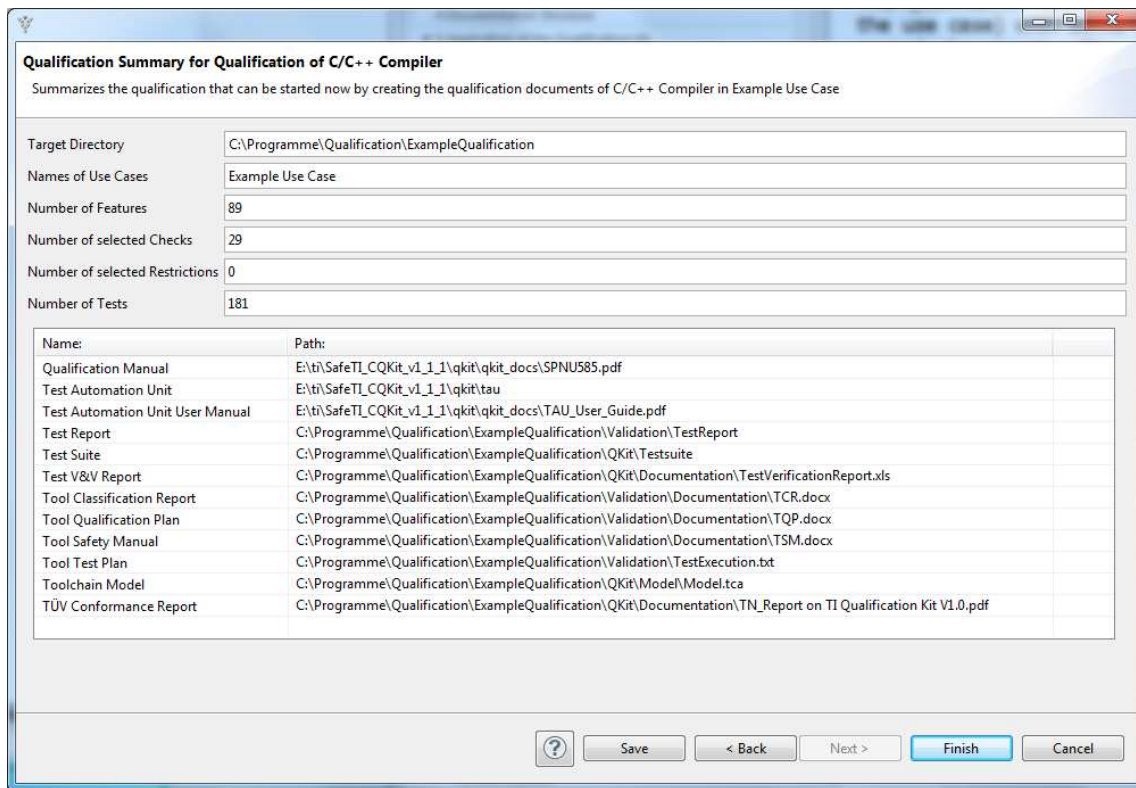
The qualification document includes the following documents depending on the selected model:

- Tool Classification Report
- Tool Qualification Plan (if tests need to be executed)
- Tool Safety Manual
- Test Plan (if tests need to be executed)

The other documents mentioned in [Chapter 4](#) do not depend on the method. The verification and verification report of the test cases are currently not generated for this qualification kit, since it is still in work. Therefore all available test cases will be executed.

NOTE: Do not forget to press the Save button if you want to continue with your selection, for example, to document your results using QST.

The qualification tool shows a summary page (after the configuration of the use case) with some statistics and the paths of the generated and copied documents. The paths are all subdirectories of the chosen qualification directory. [Figure 5-20](#) shows an example:



Qualification Summary for Qualification of C/C++ Compiler
Summarizes the qualification that can be started now by creating the qualification documents of C/C++ Compiler in Example Use Case

Target Directory: C:\Programme\Qualification\ExampleQualification

Names of Use Cases: Example Use Case

Number of Features: 89

Number of selected Checks: 29

Number of selected Restrictions: 0

Number of Tests: 181

Name:	Path:
Qualification Manual	E:\ti\SafeTI_CQKit_v1_1_1\qkit\qkit_docs\SPNU585.pdf
Test Automation Unit	E:\ti\SafeTI_CQKit_v1_1_1\qkit\tau
Test Automation Unit User Manual	E:\ti\SafeTI_CQKit_v1_1_1\qkit\qkit_docs\TAU_User_Guide.pdf
Test Report	C:\Programme\Qualification\ExampleQualification\Validation\TestReport
Test Suite	C:\Programme\Qualification\ExampleQualification\QKit\Testsuite
Test V&V Report	C:\Programme\Qualification\ExampleQualification\QKit\Documentation\TestVerificationReport.xls
Tool Classification Report	C:\Programme\Qualification\ExampleQualification\Validation\Documentation\TCR.docx
Tool Qualification Plan	C:\Programme\Qualification\ExampleQualification\Validation\Documentation\TQP.docx
Tool Safety Manual	C:\Programme\Qualification\ExampleQualification\Validation\Documentation\TSM.docx
Tool Test Plan	C:\Programme\Qualification\ExampleQualification\Validation\TestExecution.txt
Toolchain Model	C:\Programme\Qualification\ExampleQualification\QKit\Model\Model.tca
TÜV Conformance Report	C:\Programme\Qualification\ExampleQualification\QKit\Documentation\TN_Report on TI Qualification Kit V1.0.pdf

Buttons: ? Save < Back Next > Finish Cancel

Figure 5-20. Qualification Summary with Paths to Generated Documents

Press the “Finish” button to begin generation of the documents.

When the generation of documents is complete, the QST shows as depicted in [Figure 5-21](#).

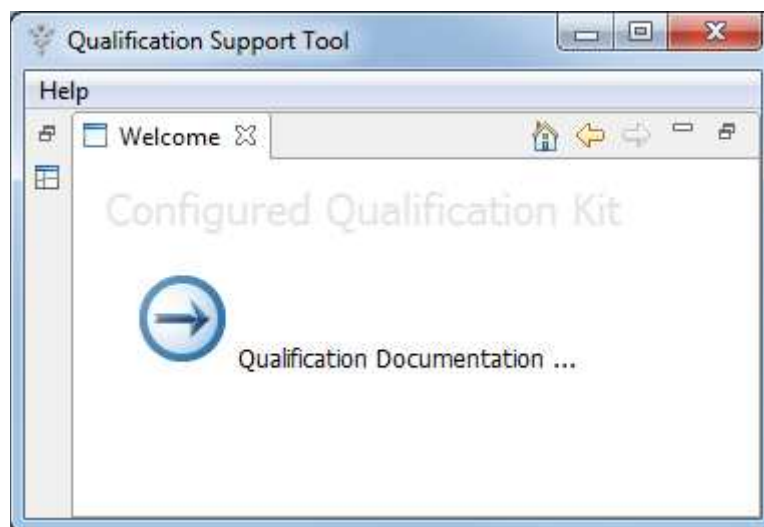


Figure 5-21. Finished Qualification Message

Click the Qualification Documentation link; the overview page of the material is shown in [Figure 5-22](#). The documents can be opened by clicking on them (or via the file system).

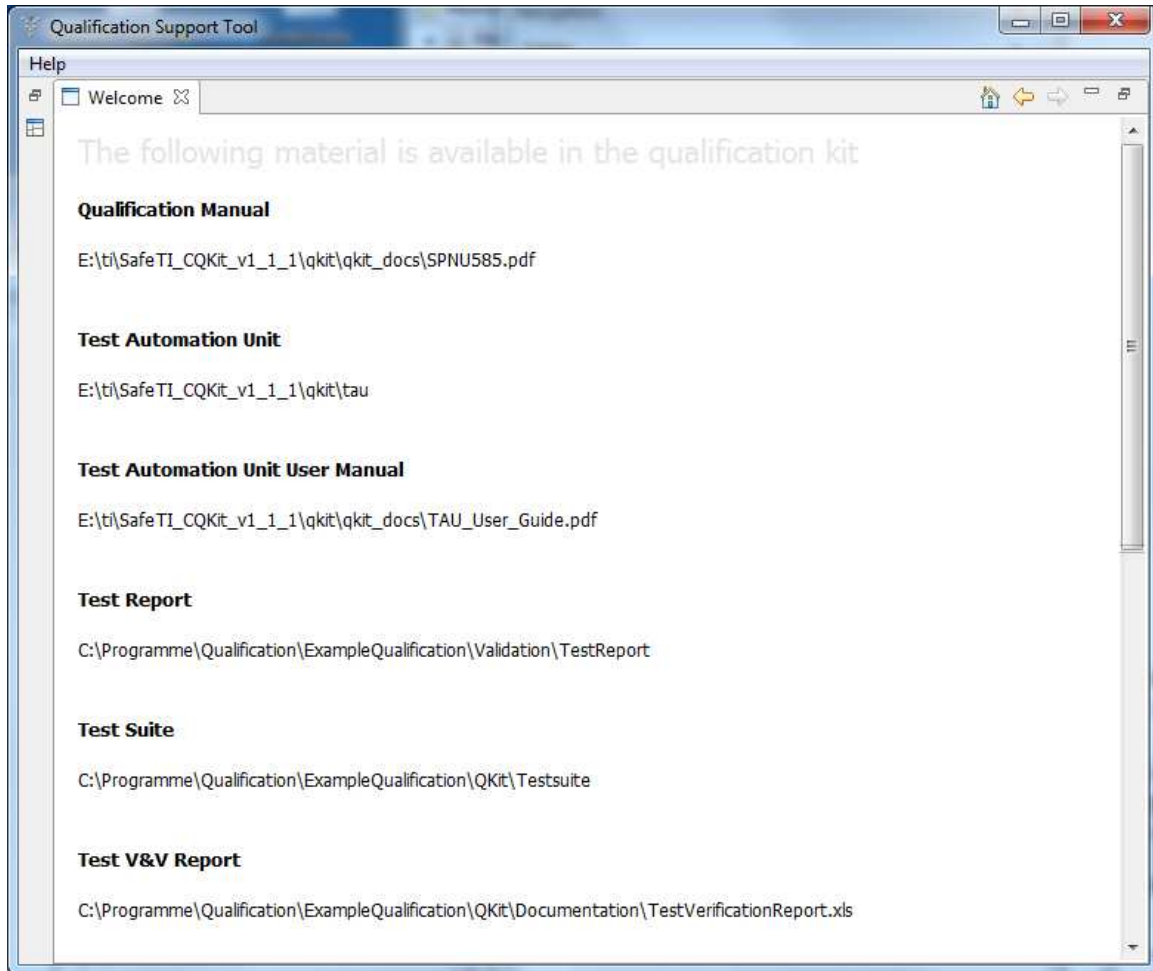


Figure 5-22. Generated Documents Overview

After the documents have been generated, they need to be updated, since they have project-specific contents (configuration). For that purpose the documents contain a document variable “client” that can be replaced by the company or project. Furthermore, the version of the document needs to be updated after the document has been reviewed. [Figure 5-23](#) shows the setting dialog of Word (German version) that allows you to change the user/project name in the generated documents.

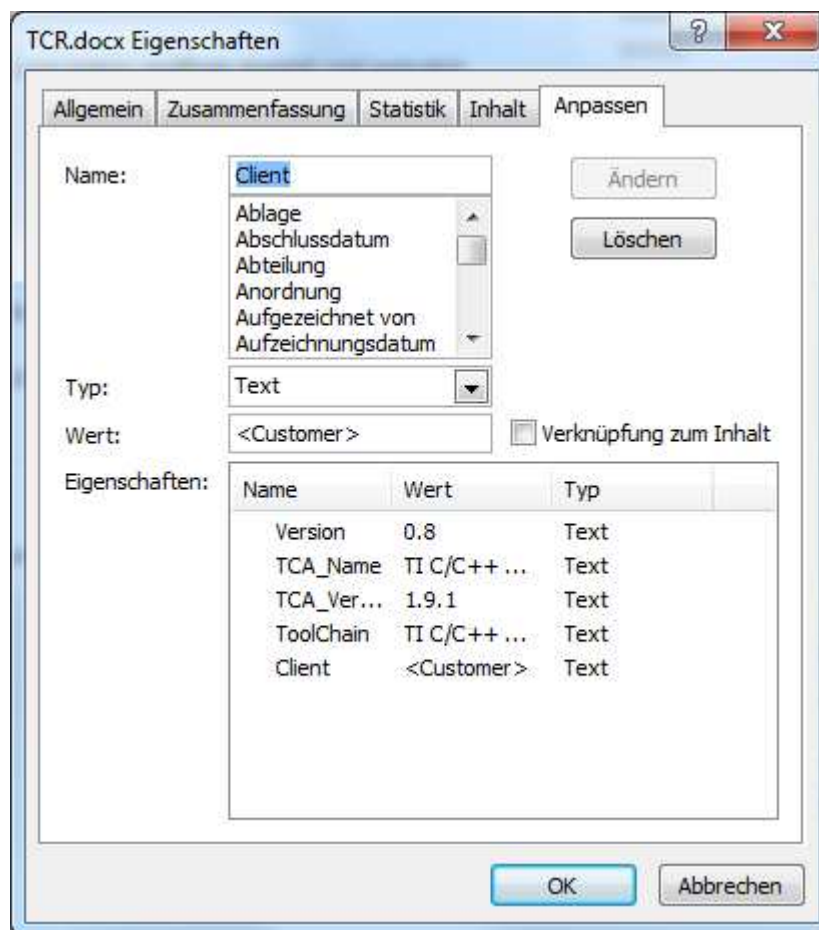


Figure 5-23. Customization of Document Properties

After changing the required properties of the documents that need to be updated, select everything (Strg+A) and update it with the F9 key.

5.3.7 Finalization of the Tool Qualification Report

The tool qualification report is built by renaming and extending the tool qualification plan. The test and qualification results should be added and the qualification process is documented there.

- The tool qualification plan contains template chapters that need only to be adapted for the results. The process can be documented in the generated section with the planned steps and documents. This can be either done manually (by directly editing the document) or using the QST (by adapting the model and regenerating the document).
- In case the QST is used, the QST has to be started and the qualification directory has to be chosen. In the model, all planning information is stored (planned dates..) and can be updated according to the real dates and steps.
- Regeneration of the TQP will add all new information into the report.
- The last step is to rename the extended and updated tool qualification plan to the tool qualification report and to review it.

5.3.8 Customization

The customization of the qualification kit can be done in two places:

- The qualification kit: for all qualifications
- The qualification directory: for the concrete qualification

The qualification support tool copies the qualification material from the qualification kit to the qualification directory and then uses it.

The qualification kit (QKit) is stored in the tool directory within a subdirectory (QKit) within a subdirectory called <plugins/MyModel> (see [Figure 5-24](#)). This contains a documentation directory with the documentation and the templates.

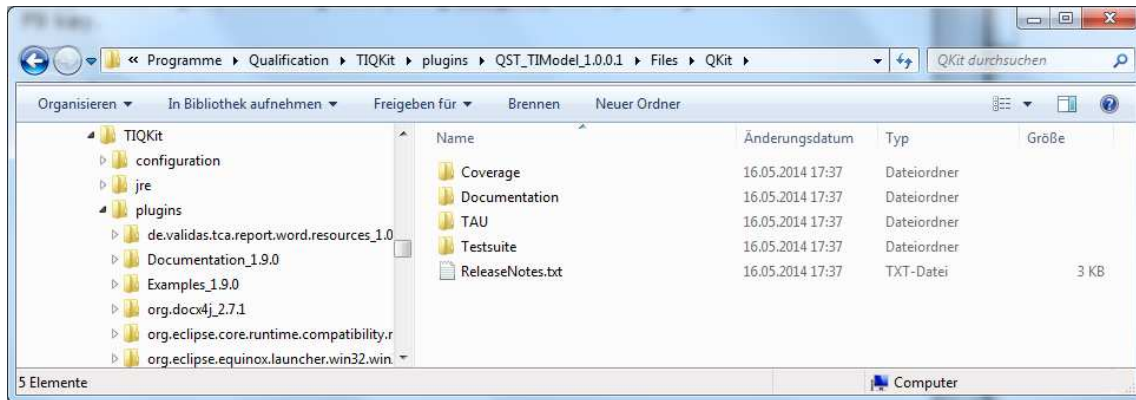


Figure 5-24. Qualifications Kit in the QST

After the qualification, the documents are copied to the qualification target directory. This has a similar structure and specifically contains the QKit (see [Figure 5-25](#)).

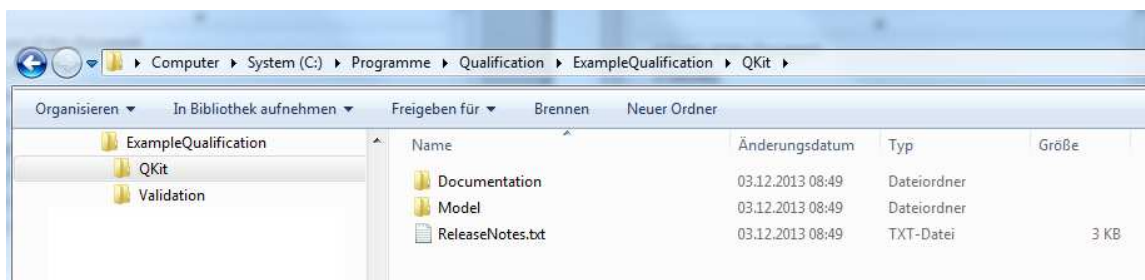


Figure 5-25. Qualification Kit in the Qualification Target Directory

Note that documents that already exist in the qualification target are not copied again. Specifically, if the qualification kit is changed, the qualification target has to be cleared so that it will not work with the old copies of the templates.

The qualification support tool creates a model of the specified qualification configuration and stores it in the qualification target directory. The stored model can be used to integrate the TI C/C++ Compiler into a development tool chain modeled with the TCA tool, or it can be used as a starting point for further qualifications.

Extension of the Qualification Kit

For the extension of the qualification kit, please see the Tool Chain Analyzer (TCA) User Guide.

References

- [DO330] RTCA. DO-330: Software Tool Qualification Considerations 1st Edition 2011-12-13.
- [EN50128]: BS EN 50128:2011, Railway applications — Communication, signaling and processing systems — Software for railway control and protection systems, BSI Standards Publication
- [IEC61508] International Electrotechnical Commission, IEC 61508, functional safety of electrical/electronic/programmable electronic safety-related systems, Edition 2.0, Apr 2010.
- [ISO26262] International Organization for Standardization. ISO 26262 Road Vehicles –Functional safety—. 1st Edition, 2011-11-15.
- [Model] The qualification model for the TI C/C++ Compiler. It is contained in the qualification kit and can be opened and changed using the [TCA].
- [SAFECOMP12] Determining Potential Errors in Tool Chains: Strategies to Reach Tool Confidence According to ISO 26262, SAFECOMP 2012, Wildmoser, Philipps, Slotosch
- [TAG] Tool Application Guide (Safety Manual) for TI C/C++ Compiler
- [TAU_UG] Test Automation Unit for TI C/C++ Compiler, contained in this qualification kit contained in the documentation of this kit in the file: TAU_User_Guide.pdf
- [TCA] Tool Chain Analyzer, tool available on www.validas.de/TCA.html Version 1.8.2dev
- [TCA_UM] Tool Chain Analyzer, Version 1.8.2dev, User Manual, (<TCAHome>/plugins/Documentation/UserManual.pdf)
- [TCR] Tool Classification Report for TI C/C++ Compiler
- [TQP] Tool Qualification Plan for TI C/C++ Compiler
- [TQR] Tool Qualification Report for TI C/C++ Compiler

Requirements Tracing to Safety Standards

The requirements of the safety standards relevant for tool qualification and safety guidelines that are considered here mainly come from the standards ISO 26262, IEC 61508, EN50128 and DO-330. Many other safety standards DO-178-C, and so forth) have similar requirements. The relevant requirements are listed (as publishable) within this section and their tracing into the following sections of this manual (or to related documents) is given.

A.1 Requirements of ISO 26262

The requirements for tools in the ISO 26262 are distributed in several parts. Figure A-1 shows the relations between the ISO 26262 requirements for tool qualification.

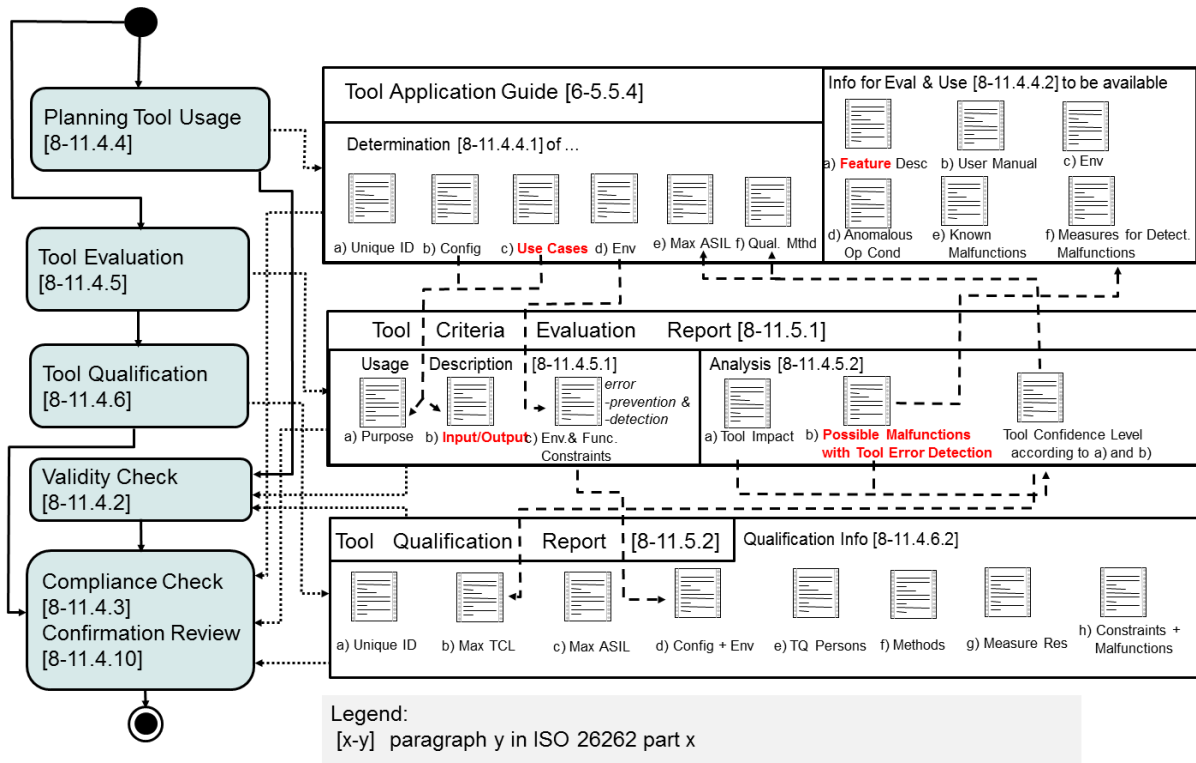


Figure A-1. ISO 26262 Tool Qualification Requirements

The following requirements are stated in the ISO 26262: ⁽²⁾

- Part 6. Chapter 5: Initiation of product development at the software level
 - [ISO_6_5_45] for each sub-phase of software development select methods and tools including guidelines for their application.
 - [ISO_6_5_46a] unambiguous definition.
 - [ISO_6_5_54] **Tool application guidelines** resulting from requirements 5.4.5 and 5.4.6.

⁽²⁾ The confirmation was obtained from ISO 26262 to list the following requirements for better demonstration of the conformance.

- Part 8. Chapter 11: Confidence in the use of software tools contains the analysis and the qualification methods
 - [ISO_8_11_42] Validity of predetermined tool confidence level or qualification
 - [ISO_8_11_421] If the confidence level evaluation or qualification of a software tool is performed independently from the development of a particular safety-related item or element, the validity of this predetermined tool confidence level or qualification will be confirmed, in accordance with ISO 26262-2:2011, prior to the software tool being used for the development of a particular safety-related item or element
 - [ISO_8_11_431] When using a software tool, it will be ensured that its usage, its determined environmental and functional constraints and its general operating conditions comply with its evaluation criteria or its qualification.
 - [ISO_8_11_441] The usage of a software tool will be planned, including the determination of:
 - [ISO_8_11_441a] the identification and version number of the software tool
 - [ISO_8_11_441b] the configuration of the software tool
 - [ISO_8_11_441c] the use cases of the software tool
 - [ISO_8_11_441d] the environment in which the software tool is executed
 - [ISO_8_11_441e] the maximum ASIL of all the safety requirements, allocated to the item or the element that can be violated, if the software tool is malfunctioning and producing corresponding erroneous output
 - [ISO_8_11_441f] the measures for the detection of malfunctions and the corresponding erroneous output of the software tool identified during the determination of the required level of confidence for this software tool
 - [ISO_8_11_442] To ensure the proper evaluation or usage of the software tool, the following information will be available:
 - [ISO_8_11_442a] description of the features, functions and technical properties of the software tool
 - [ISO_8_11_442b] the user manual or other usage guides, if applicable
 - [ISO_8_11_442c] a description of the environment required for its operation
 - [ISO_8_11_442d] a description of the expected behavior of the software tool under anomalous operating conditions, if applicable
 - [ISO_8_11_442e] a description of known software tool malfunctions and the appropriate safeguards, avoidance or work-around measures, if applicable,
 - [ISO_8_11_442f] the measures for the detection of malfunctions and the corresponding erroneous output of the software tool identified during the determination of the required level of confidence for this software tool
 - [ISO_8_11_451] The description of the usage of a software tool will contain the following information:
 - [ISO_8_11_451a] the intended purpose
 - [ISO_8_11_451b] the inputs and expected outputs, and
 - [ISO_8_11_451c] the environmental and functional constraints, if applicable
 - [ISO_8_11_452] The intended usage of the software tool will be analyzed and evaluated to determine:
 - [ISO_8_11_452a] the possibility that a malfunction of a particular software tool can introduce or fail to detect errors in a safety-related item or element being developed. This is expressed by the classes of Tool Impact
 - [ISO_8_11_452b] the confidence in measures that prevent the software tool from malfunctioning and producing corresponding erroneous output, or in measures that detect that the software tool has malfunctioned and has produced corresponding erroneous output. This is expressed by the classes of Tool error Detection (TD):
 - TD1 will be selected if there is a high degree of confidence that a malfunction and its corresponding erroneous output will be prevented or detected

- TD2 will be selected if there is a medium degree of confidence that a malfunction and its corresponding erroneous output will be prevented or detected
- TD3 will be selected in all other cases
- [ISO_8_11_462] The qualification of the software tool will be documented including the following:
 - [ISO_8_11_462a] the unique identification and version number of the software tool
 - [ISO_8_11_462b] the maximum Tool Confidence Level for which the software tool is classified together with a reference to its evaluation analysis
 - [ISO_8_11_462c] the pre-determined maximum ASIL, or specific ASIL, of any safety requirement which might be violated if the software tool is malfunctioning and produces corresponding erroneous output
 - [ISO_8_11_462d] the configuration and environment for which the software tool is qualified
 - [ISO_8_11_462e] the person or organization who carried out the qualification
 - [ISO_8_11_462f] the methods applied for its qualification in accordance with 11.4.6.1
 - [ISO_8_11_462g] the results of the measures applied to qualify the software tool
 - [ISO_8_11_462h] the usage constraints and malfunctions identified during the qualification, if applicable
- o [ISO_8_11_410] Confirmation review of qualification of a software tool. This subclause applies to ASILs (B), C, D, in accordance with 4.3. The confidence in the use of the software tool will be evaluated in accordance with ISO 26262-2:2011 Table 1 to ensure:
 - [ISO_8_11_410a] the correct evaluation of the required level of confidence in the software tool
 - [ISO_8_11_410b] the appropriate qualification of the software tool in accordance with its required level of confidence

A.2 Requirements of IEC 61508

The requirements for tools in the IEC 61508 are distributed in several parts. Part 4 contains the relevant tool definitions for the classes T2 (test tools) and T3 (constructive tools), while Part 3 contains software requirements. The commitment to list the tool qualification requirement was not obtained, but the numbering scheme is identical to the standard.

A.3 Requirements of EN 50128

In the *Support Tools and Languages* section of [EN50128], the following requirements are described. Many of them are covered by the tool qualification (see [Section 5.3](#)) and have a VS-EN-ID. The others are argued to be not applicable for this qualification and will be covered by the surrounding safety process.

A.4 Requirements of DO-330 (Operation Parts)

The DO-330 is safety standard for the development of tools and therefore only the parts for the determination of the required confidence (classification) and the tool operational requirements are considered. The used notions of Tool operational requirements (TORs) and Tool Requirements (TRs) correspond to our terms use case and feature. The concrete list has been removed since the permission to list them has just been obtained.

A.5 Satisfaction of ISO 26262 Requirements

The identified requirements from [Section A.1](#) are satisfied as follows:

- [ISO_6_5_45]: this document contains an unique identification of the TI C/C++ Compiler ([Section 5.1](#)) and the guidelines how to use it
- [ISO_6_5_46a]: see the definition of the TI C/C++ Compiler in Section 5.1 in [TAG]
- [ISO_6_5_54]: see the guidelines how to apply the tool in Sections 5, 6,7 in [TAG]
- [ISO_8_11_42]: see TG_TCA_52_2 in Section 5.1 in [TAG]
- [ISO_8_11_421]: see TG_TCA_52_2 in Section 5.1 in [TAG] (the TCL has been determined in [TCR])
- [ISO_8_11_431]: see Sections 5, 6, and 7 in [TAG]

- [ISO_8_11_441]: see tracing of [ISO_8_11_441a] to [ISO_8_11_441f]
- [ISO_8_11_441a]: see Section 5.1 in [TAG]
- [ISO_8_11_441b]: see Section 5.1 in [TAG]
- [ISO_8_11_441c]: see Section 6 in [TAG]
- [ISO_8_11_441d]: see Section 5.1 in [TAG]
- [ISO_8_11_441e]: see TG_TCA_52_4 in Section 5.1 in [TAG]
- [ISO_8_11_441f]: see the results in [TCR]
- [ISO_8_11_442]: see tracing of [ISO_8_11_442a] to [ISO_8_11_442f]
- [ISO_8_11_442a]: see [TCR] and Section 5.3 in [TAG]
- [ISO_8_11_442b]: see Section 5.1 in [TAG]
- [ISO_8_11_442b]: see Section 5.1 in [TAG]
- [ISO_8_11_442d]: see Section 7.3 in [TAG]
- [ISO_8_11_442e]: see Section 5.4.2 in [TAG]
- [ISO_8_11_442f]: see Sections 7.1 and 7.2 in [TAG]
- [ISO_8_11_451]: see tracing of [ISO_8_11_451a] to [ISO_8_11_451c]
- [ISO_8_11_451a]: see the description of use cases in Section 6 in [TAG]
- [ISO_8_11_451b]: see the description of features and use cases in Sections 5 and 6 in [TAG]
- [ISO_8_11_451b]: see the description of features and use cases in Sections 5 and 6 in [TAG]
- [ISO_8_11_452], [ISO_8_11_452a], [ISO_8_11_452b]: has been determined in [TCR]
- [ISO_8_11_462]: see tracing of [ISO_8_11_462a] to [ISO_8_11_462h]
- [ISO_8_11_462a]: see Section 5.1 in [TAG]
- [ISO_8_11_462b]: see TG_TCA_52_2 and in [TCR]
- [ISO_8_11_462c]: see TG_TCA_52_4 in Section 5.2 in [TAG]
- [ISO_8_11_462d]: see TG_TCA_52_3 in Section 5.2, TG_TCA_53_2 in Section 5.3, and UG_TCA_61_3 in Section 6.1 in [TAG]
- [ISO_8_11_462e]: see TG_TCA_52_3 in Section 5.2, TG_TCA_53_2 in Section 5.3, and UG_TCA_61_3 in Section 6.1 in [TAG]
- [ISO_8_11_462f]: see TG_TCA_52_3 in Section 5.2, TG_TCA_53_2 in Section 5.3, and UG_TCA_61_3 in Section 6.1 in [TAG]
- [ISO_8_11_462g]: see TG_TCA_54_5 in Section 5.4 in [TAG]
- [ISO_8_11_462h]: see TG_TCA_7_5 in Section 7 in [TAG]
- [ISO_8_11_410]: see [ISO_8_11_410a] and [ISO_8_11_410b]
- [ISO_8_11_410a]: see TG_TCA_52_1 in Section 5.2 in [TAG]
- [ISO_8_11_410b]: see TG_TCA_52_2 in Section 5.2 in [TAG]

A.6 Satisfaction of IEC-61508 Requirements

The identified requirements from [Section A.2](#) are satisfied as follows:

- [IEC_3_7443] is done in [TCR]
- [IEC_3_7444] qualification needs are satisfied by qualification, which is verified in Section 5.2 in [TAG]
- [IEC_3_7445] is done for all off-line tools [TCR]
- [IEC_3_7446] tool qualification provides the evidence and is checked in Section 5.2 in [TAG]
- [IEC_3_7447] see [IEC_3_7447a] to [IEC_3_7447g]
- [IEC_3_7447a] see qualification report of the tool and the checks in TG_TCA_52_3, TG_TCA_53_2 and UG_TCA_61_3
- [IEC_3_7447b] see Section 5.1
- [IEC_3_7447c] see Section 5.3

- [IEC_3_7447d] see tool qualification plan
- [IEC_3_7447e] see qualification report
- [IEC_3_7447f] part of qualification report
- [IEC_3_7447g] part of qualification report
- [IEC_3_7448] proposed in tools [TCR] and validated by confirmation review in TG_TCA_52_3, TG_TCA_53_2 and UG_TCA_61_3
- [IEC_3_7449] is analyzed by the artifact flow in [TCR]
- [IEC_3_74415] satisfied by [IEC_3_74415a] to [IEC_3_74415c]
- [IEC_3_74415a] see Section 5.1
- [IEC_3_74415b] see TG_TCA_51_3
- [IEC_3_74415c] see use case description in Section 6.1 and UG_TCA_61_1
- [IEC_3_74416] see TG_TCA_52_5, other tools are TCL 1
- [IEC_3_74417] is analyzed by the artifact flow in [TCR]
- [IEC_3_74418] is implied by TG_TCA_52_5
- [IEC_3_74418a] is implied by TG_TCA_52_5 and TG_TCA_7_5
- [IEC_3_74418b] reuse qualification tests in tool qualification plan
- [IEC_4_3211] definitions (see [IEC_4_3211a] until [IEC_4_3211b])
- [IEC_4_3211a] T1 has no impact and is therefore TCL1
- [IEC_4_3211b] T2 can oversee errors and is therefore TCL2 or TCL3
- [IEC_4_3211c] T3 can introduce errors and is therefore TCL2 or TCL3

A.7 Satisfaction of EN 50128

The requirements from [Section A.3](#) for T3 tools for are covered as follows:

- VS-EN-11: The tool qualification report which extends this tool qualification plan documents the performed activities.
- VS-EN-12: The version of the tool is contained in Section 5 of [TQP] and the [TAG].
- VS-EN-13: The validated tool functions are modeled as Features and are listed in Section 5 of [TQP].
- VS-EN-14: The TAU is described in [TAU_UG] that is referred to in this qualification plan and the resulting qualification report.
- VS-EN-15: The tool qualification report which extends this tool qualification plan documents the result of the validation.
- VS-EN-16: The test report which is generated from the TAU by processing the test plan contains the test results.
- VS-EN-17: The test report contains also the discrepancies (failed tests).
- VS-EN-18: [TCR] and [TAG] contain effective measures for features that are not qualified according to this qualification plan.
- VS-EN-20: The qualification plan for the TI C/C++ Compiler [TQP] satisfies international standards.
- VS-EN-21: The matching to the application is achieved by selecting the required features of the tool and verified by comparing the code coverage between the qualification and the application of the TI C/C++ Compiler. This is documented in [TCR] and [TAG].
- VS-EN-23: The required features have been selected from the user to match the development process and are listed in Section 5 of [TQP].
- VS-EN-25: The qualification in [TQP] is project specific by selecting the features and comparing the code coverage in the TI C/C++ Compiler during qualification with the application.
- VS-EN-26: The selected qualification method is validation of the selected features and contains functional tests (see Section 6.5.2 of [TQP]) as well as robustness tests (see Section 6.5.3 of [TQP]).

- VS-EN-31: The tool qualification plan [TQP] captures all relevant requirements of the EN50128 for T3 tools. For T1 and T2 tools no qualification is required, but an analysis of potential errors as was done during the creation of the qualification kit. This is configured from the user during application of the kit and documents the relevant analysis in [TCR] and [TAG] that justify the selection of the TI C/C++ Compiler.

A.8 Satisfaction of Requirements of DO-330 (Operational Parts)

The identified requirements from [Section A.3](#) are satisfied as follows:

- [DO_330_41] see [DO_330_41a] until [DO_330_41f]
- [DO_330_41] see [DO_330_41a] until [DO_330_41f]
- [DO_330_41b] see use cases in Section 6.1
- [DO_330_41c] is done in [TCR]
- [DO_330_41d] is done in [TCR] and can be mapped to TQLs
- [DO_330_41e] is done in tool qualification plan
- [DO_330_41f] see section 5.1
- [DO_330_1011]: see [DO_330_1011a] to [DO_330_1011h]
- [DO_330_1011a] see section 5.1 and 6.1
- [DO_330_1011b] is done in [TCR]
- [DO_330_1011c] is done in tool qualification plan
- [DO_330_1011d] is done in [TCR] and can be mapped to TQLs
- [DO_330_1011e] see section 5.1
- [DO_330_1011f] is done in tool qualification plan
- [DO_330_1011g] see section 7
- [DO_330_1011h] see section 5.1 and 6.1
- [DO_330_1031] see [DO_330_1031a] to [DO_330_1031i]
- [DO_330_1031a] is done in [TCR] and section 5.1
- [DO_330_1031a] is done in [TCR] and section 5.1
- [DO_330_1031a] is done in [TCR] and section 5.1
- [DO_330_1031a] is done in [TCR] and section 5.1
- [DO_330_1031e] is done in [TCR] and sections 5.3 and 6.1
- [DO_330_1031f] is done in tool qualification plan.
- [DO_330_1031g] see section 5.1
- [DO_330_1031h] see use-cases in section 6.1
- [DO_330_1031i] Performance is not considered for safety relevant failures.
- [DO_330_1034] see [DO_330_1034a] until [DO_330_1034d]
 - [DO_330_1034a] is done in tool qualification plan.
 - [DO_330_1034b] is done in tool qualification plan.
 - [DO_330_1034c] is done in tool qualification report
 - [DO_330_1034d] is done in tool qualification report
- [DO_330_D2] is done in [TCR].

Revision History

This document has been revised to accommodate the following change(s)

Table B-1. Revisions

Location	Modifications / Additions / Deletions
Chapter 5	Revised the figures in this section for minor text changes and for better clarity.

IMPORTANT NOTICE

Texas Instruments Incorporated and its subsidiaries (TI) reserve the right to make corrections, enhancements, improvements and other changes to its semiconductor products and services per JESD46, latest issue, and to discontinue any product or service per JESD48, latest issue. Buyers should obtain the latest relevant information before placing orders and should verify that such information is current and complete. All semiconductor products (also referred to herein as "components") are sold subject to TI's terms and conditions of sale supplied at the time of order acknowledgment.

TI warrants performance of its components to the specifications applicable at the time of sale, in accordance with the warranty in TI's terms and conditions of sale of semiconductor products. Testing and other quality control techniques are used to the extent TI deems necessary to support this warranty. Except where mandated by applicable law, testing of all parameters of each component is not necessarily performed.

TI assumes no liability for applications assistance or the design of Buyers' products. Buyers are responsible for their products and applications using TI components. To minimize the risks associated with Buyers' products and applications, Buyers should provide adequate design and operating safeguards.

TI does not warrant or represent that any license, either express or implied, is granted under any patent right, copyright, mask work right, or other intellectual property right relating to any combination, machine, or process in which TI components or services are used. Information published by TI regarding third-party products or services does not constitute a license to use such products or services or a warranty or endorsement thereof. Use of such information may require a license from a third party under the patents or other intellectual property of the third party, or a license from TI under the patents or other intellectual property of TI.

Reproduction of significant portions of TI information in TI data books or data sheets is permissible only if reproduction is without alteration and is accompanied by all associated warranties, conditions, limitations, and notices. TI is not responsible or liable for such altered documentation. Information of third parties may be subject to additional restrictions.

Resale of TI components or services with statements different from or beyond the parameters stated by TI for that component or service voids all express and any implied warranties for the associated TI component or service and is an unfair and deceptive business practice. TI is not responsible or liable for any such statements.

Buyer acknowledges and agrees that it is solely responsible for compliance with all legal, regulatory and safety-related requirements concerning its products, and any use of TI components in its applications, notwithstanding any applications-related information or support that may be provided by TI. Buyer represents and agrees that it has all the necessary expertise to create and implement safeguards which anticipate dangerous consequences of failures, monitor failures and their consequences, lessen the likelihood of failures that might cause harm and take appropriate remedial actions. Buyer will fully indemnify TI and its representatives against any damages arising out of the use of any TI components in safety-critical applications.

In some cases, TI components may be promoted specifically to facilitate safety-related applications. With such components, TI's goal is to help enable customers to design and create their own end-product solutions that meet applicable functional safety standards and requirements. Nonetheless, such components are subject to these terms.

No TI components are authorized for use in FDA Class III (or similar life-critical medical equipment) unless authorized officers of the parties have executed a special agreement specifically governing such use.

Only those TI components which TI has specifically designated as military grade or "enhanced plastic" are designed and intended for use in military/aerospace applications or environments. Buyer acknowledges and agrees that any military or aerospace use of TI components which have **not** been so designated is solely at the Buyer's risk, and that Buyer is solely responsible for compliance with all legal and regulatory requirements in connection with such use.

TI has specifically designated certain components as meeting ISO/TS16949 requirements, mainly for automotive use. In any case of use of non-designated products, TI will not be responsible for any failure to meet ISO/TS16949.

Products

Audio	www.ti.com/audio
Amplifiers	amplifier.ti.com
Data Converters	dataconverter.ti.com
DLP® Products	www.dlp.com
DSP	dsp.ti.com
Clocks and Timers	www.ti.com/clocks
Interface	interface.ti.com
Logic	logic.ti.com
Power Mgmt	power.ti.com
Microcontrollers	microcontroller.ti.com
RFID	www.ti-rfid.com
OMAP Applications Processors	www.ti.com/omap
Wireless Connectivity	www.ti.com/wirelessconnectivity

Applications

Automotive and Transportation	www.ti.com/automotive
Communications and Telecom	www.ti.com/communications
Computers and Peripherals	www.ti.com/computers
Consumer Electronics	www.ti.com/consumer-apps
Energy and Lighting	www.ti.com/energy
Industrial	www.ti.com/industrial
Medical	www.ti.com/medical
Security	www.ti.com/security
Space, Avionics and Defense	www.ti.com/space-avionics-defense
Video and Imaging	www.ti.com/video

TI E2E Community

e2e.ti.com